Different versions of Inception

Analytics Vidhya

Challenges with GoogLeNet

Analytics Vidhya

Analytics Vidhya

Challenges with GoogLeNet

- High computational cost (e.g. using 5X5 or 7X7 filters)

type	patch size/ stride	output size	depth	#1×1	#3×3 reduce	#3×3	#5×5 reduce	#5×5	pool proj	params	ops
convolution	7×7/2	112×112×64	1							2.7K	34M
max pool	3×3/2	56×56×64	0								
convolution	3×3/1	56×56×192	2		64	192				112K	360M
max pool	3×3/2	28×28×192	0								
inception (3a)		28×28×256	2	64	96	128	16	32	32	159K	128M
inception (3b)		28×28×480	2	128	128	192	32	96	64	380K	304M
max pool	3×3/2	14×14×480	0								

Analytics Vidhya

Challenges with GoogLeNet

- High computational cost (e.g. using 5X5 or 7X7 filters)
- Reduce representational bottleneck

Analytics Vidhya

inception (5a)		$7 \times 7 \times 832$	2	256	160	320	32	128	128	1072K	54M
inception (5b)		$7 \times 7 \times 1024$	2	384	192	384	48	128	128	1388K	71M
avg pool	$7 \times 7/1$	$1 \times 1 \times 1024$	0								
dropout (40%)		$1 \times 1 \times 1024$	0								
linear		$1 \times 1 \times 1000$	1							1000K	1M
softmax		$1 \times 1 \times 1000$	0								

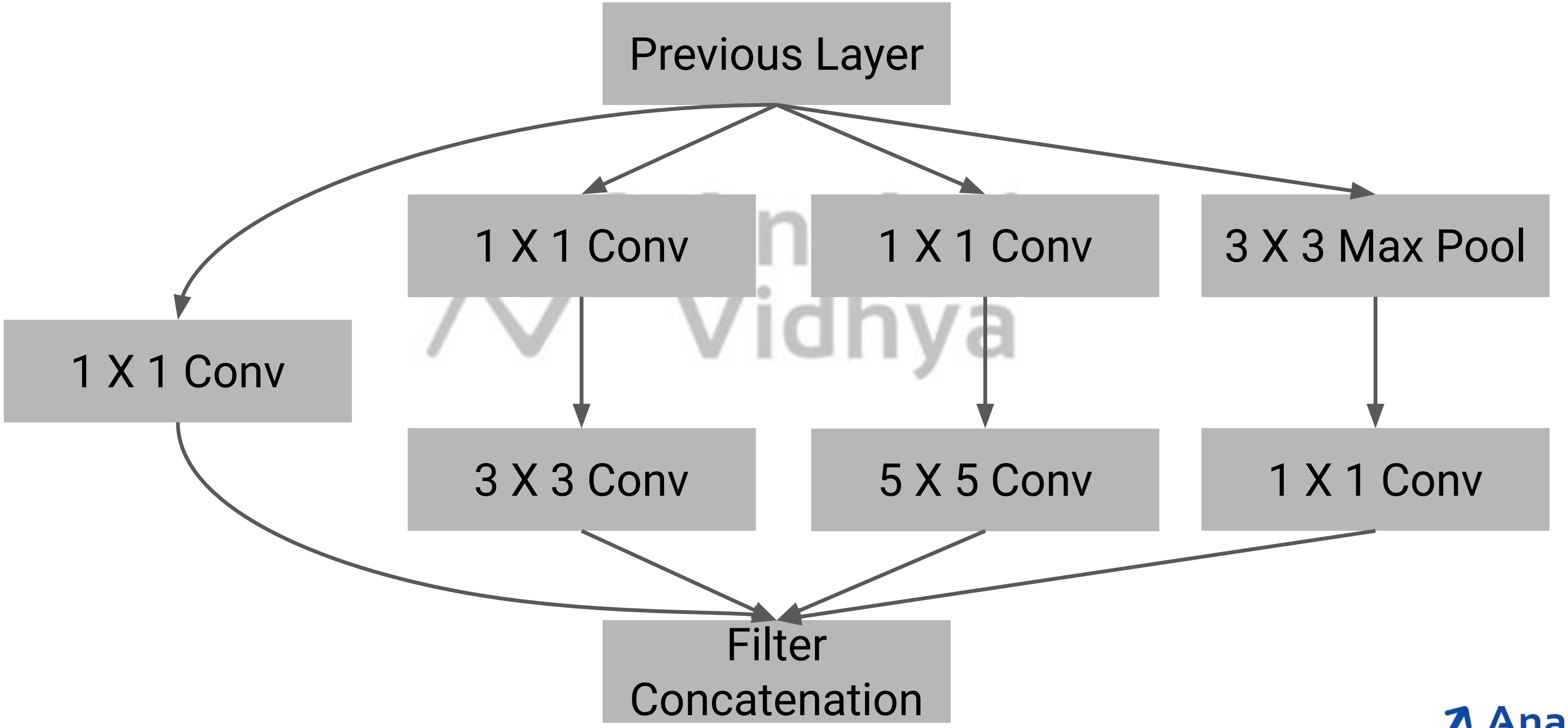
Analytics Vidhya

GoogLeNet: High Computational Cost

type	patch size/ stride	output size	depth	#1×1	#3×3 reduce	#3×3	#5×5 reduce	#5×5	pool proj	params	ops
convolution	7×7/2	112×112×64	1							2.7K	34M
max pool	3×3/2	56×56×64	0								
convolution	3×3/1	56×56×192	2		64	192				112K	360M
max pool	3×3/2	28×28×192	0								
inception (3a)		28×28×256	2	64	96	128	16	32	32	159K	128M
inception (3b)		28×28×480	2	128	128	192	32	96	64	380K	304M
max pool	3×3/2	14×14×480	0								
inception (4a)		14×14×512	2	192	96	208	16	48	64	364K	73M
inception (4b)		14×14×512	2	160	112	224	24	64	64	437K	88M
inception (4c)		14×14×512	2	128	128	256	24	64	64	463K	100M
inception (4d)		14×14×528	2	112	144	288	32	64	64	580K	119M
inception (4e)		14×14×832	2	256	160	320	32	128	128	840K	170M
max pool	3×3/2	7×7×832	0								
inception (5a)		7×7×832	2	256	160	320	32	128	128	1072K	54M
inception (5b)		7×7×1024	2	384	192	384	48	128	128	1388K	71M
avg pool	7×7/1	1×1×1024	0								
dropout (40%)		1×1×1024	0								
linear		1×1×1000	1							1000K	1M
softmax		1×1×1000	0								

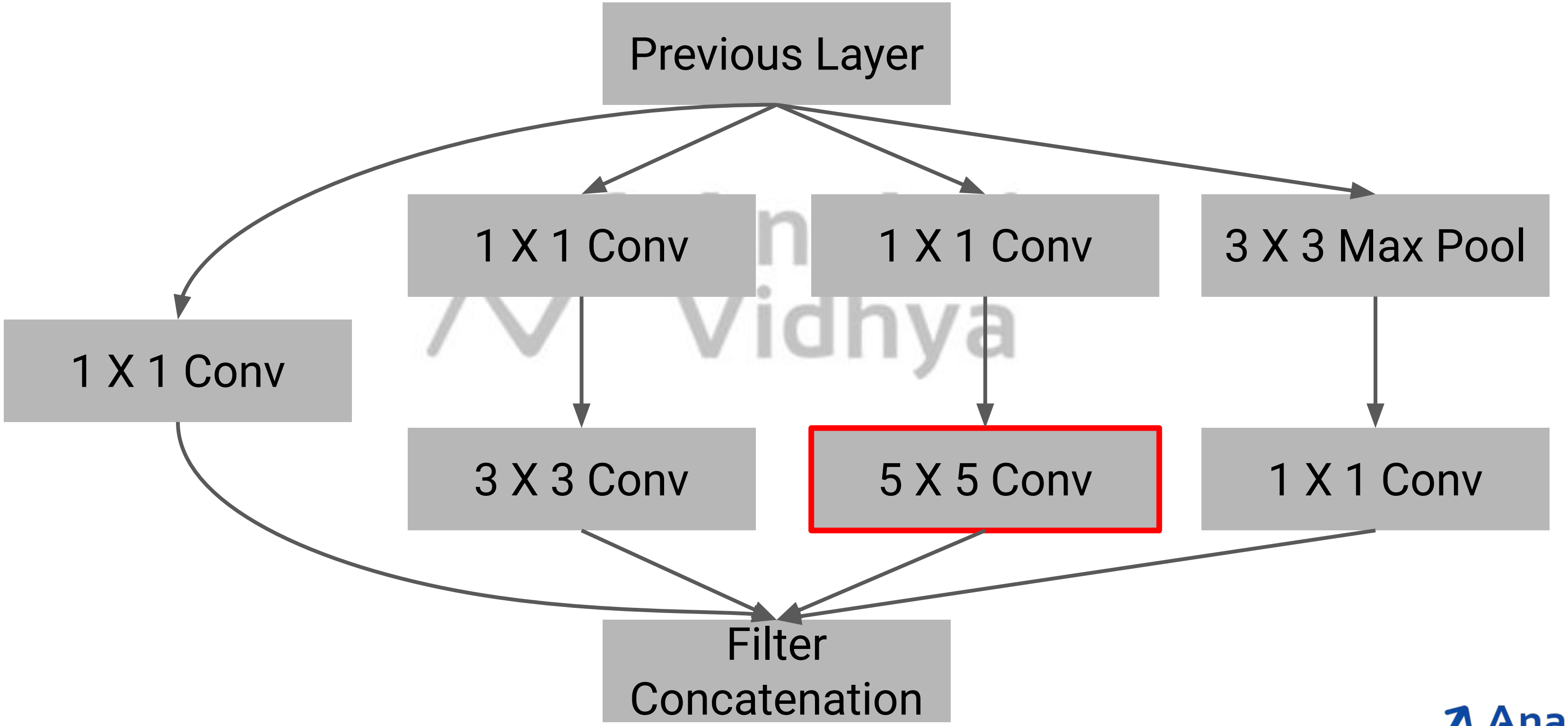
High Computational Cost
Solution: Factorized Convolution

High Computational Cost
Solution: Factorized Convolution

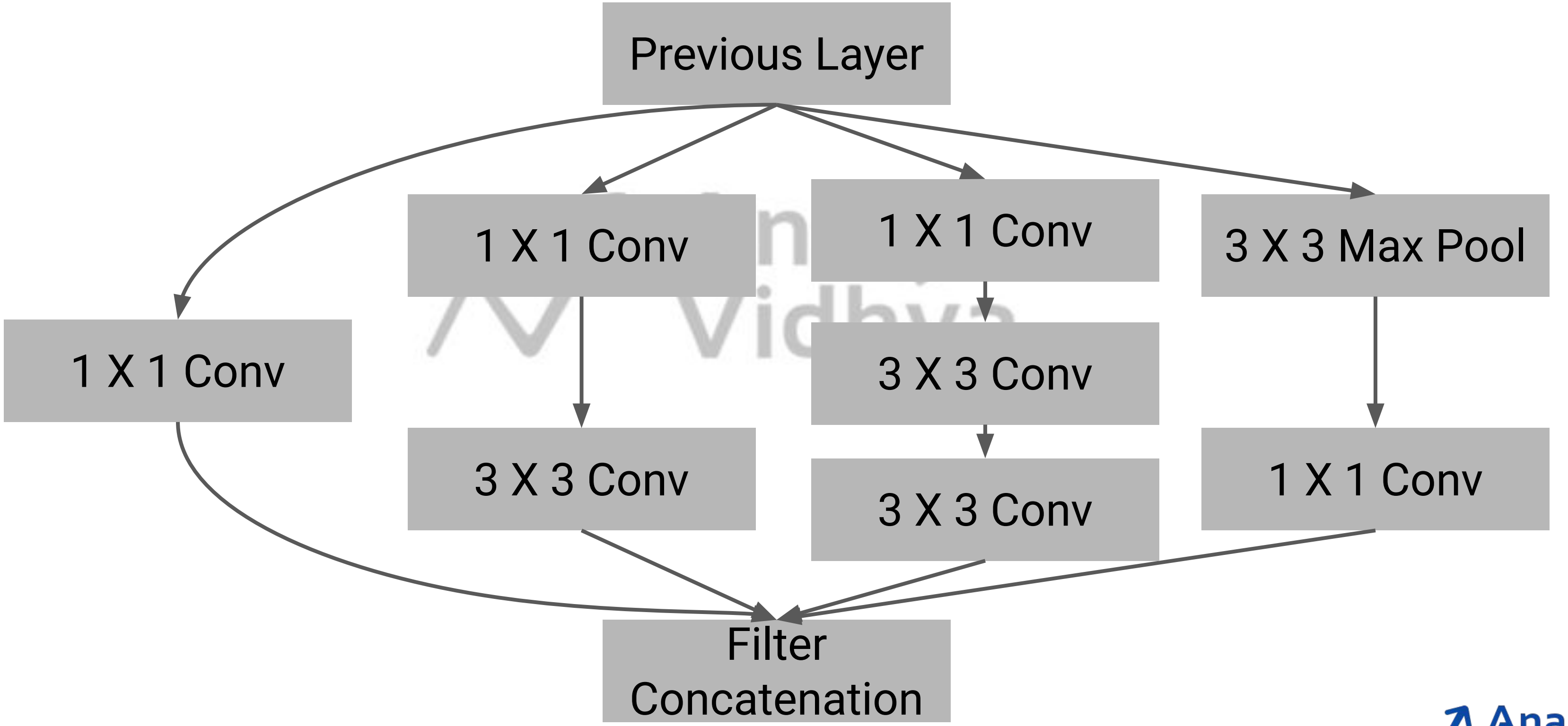


Analytics Vidhya

High Computational Cost
Solution: Factorized Convolution

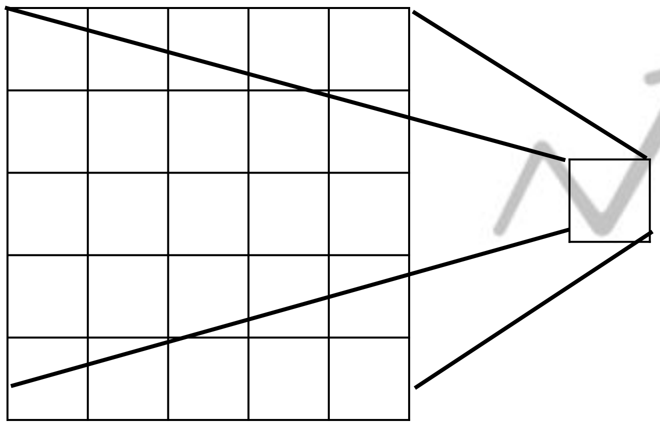


High Computational Cost
Solution: Factorized Convolution



Analytics Vidhya

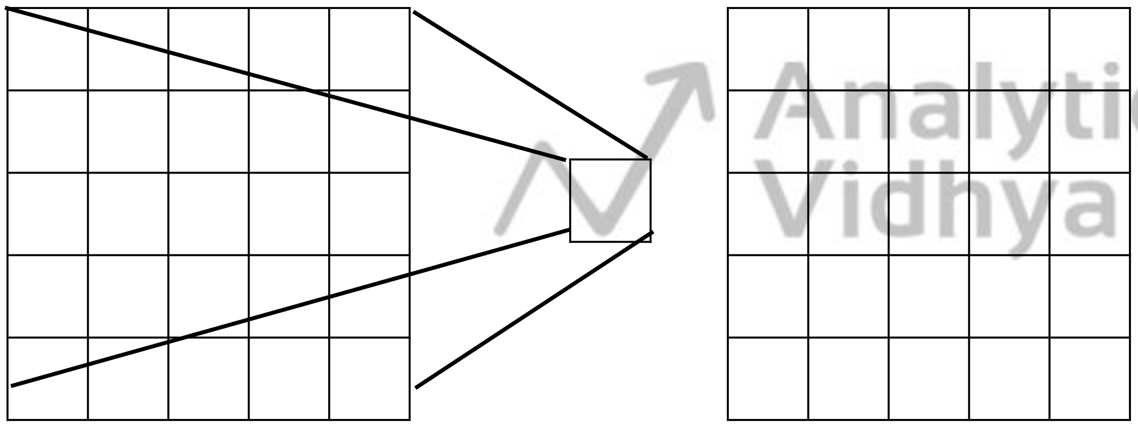
High Computational Cost
Solution: Factorized Convolution



5X5 Conv

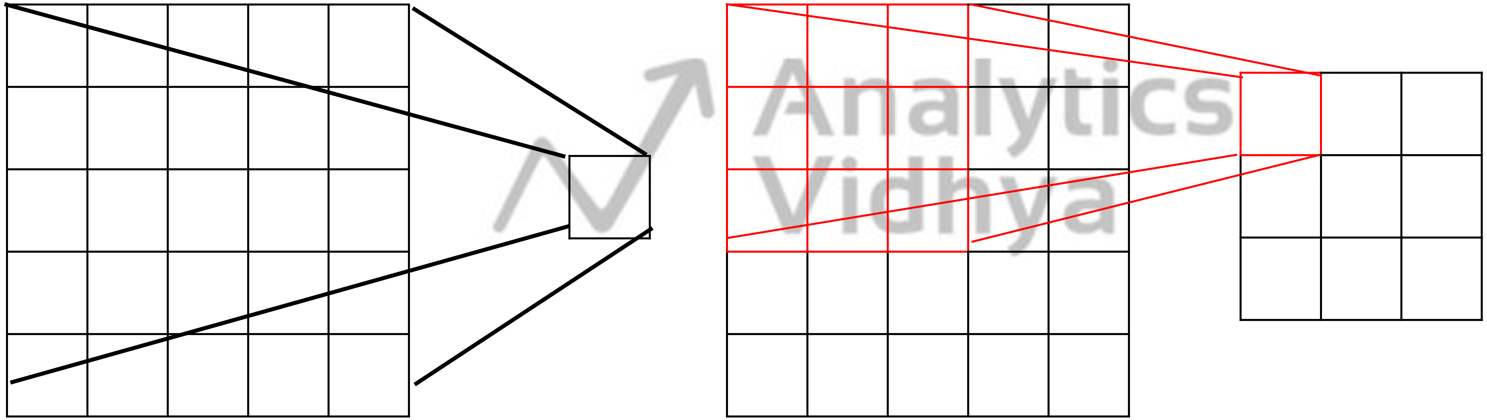
Analytics Vidhya

High Computational Cost
Solution: Factorized Convolution



5X5 Conv

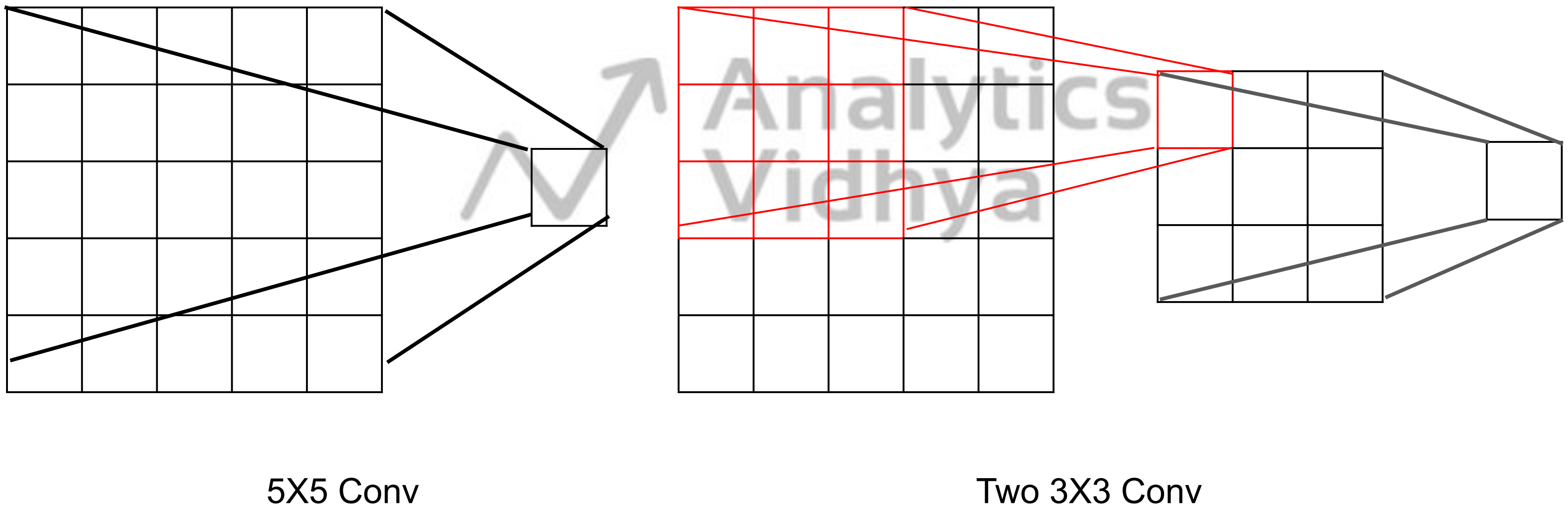
High Computational Cost
Solution: Factorized Convolution



5X5 Conv

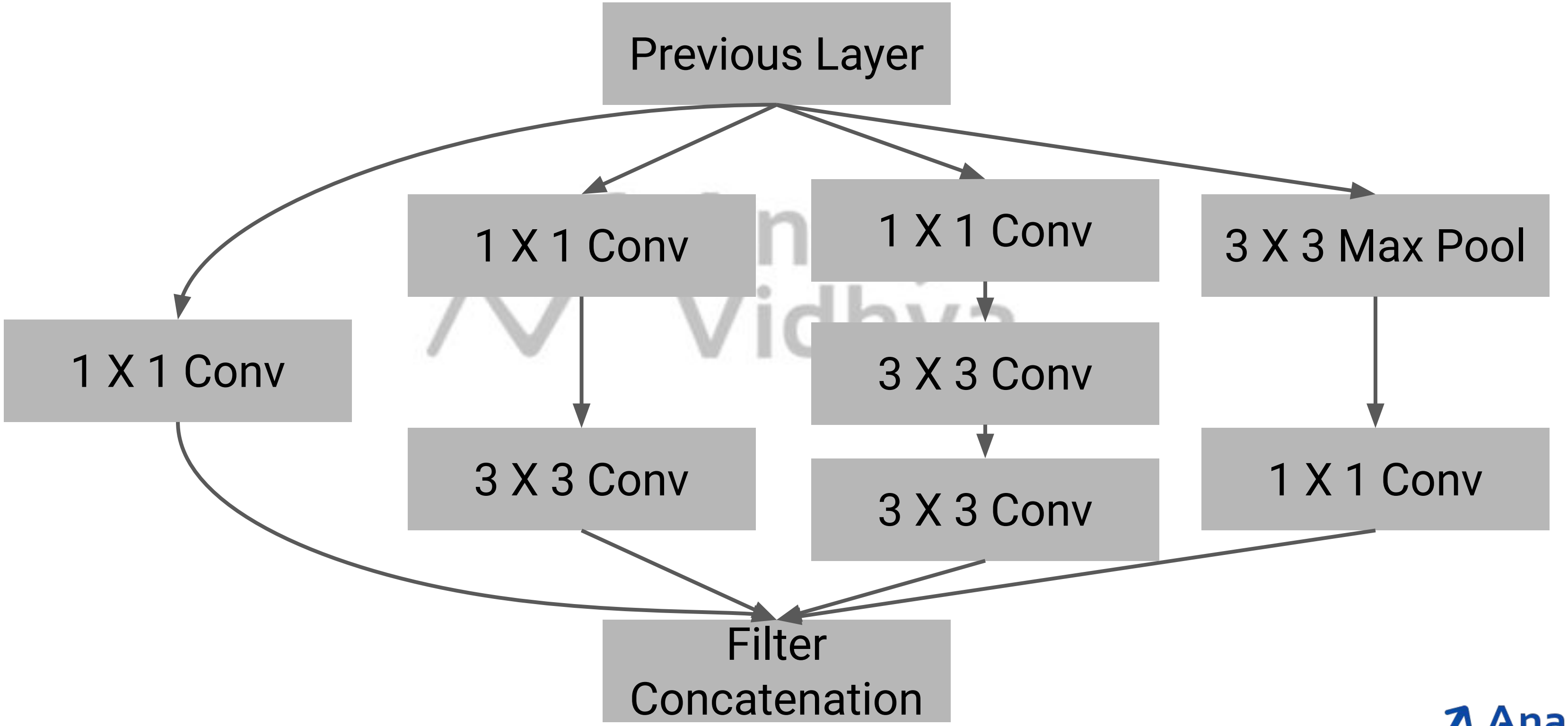
Analytics Vidhya

High Computational Cost
Solution: Factorized Convolution



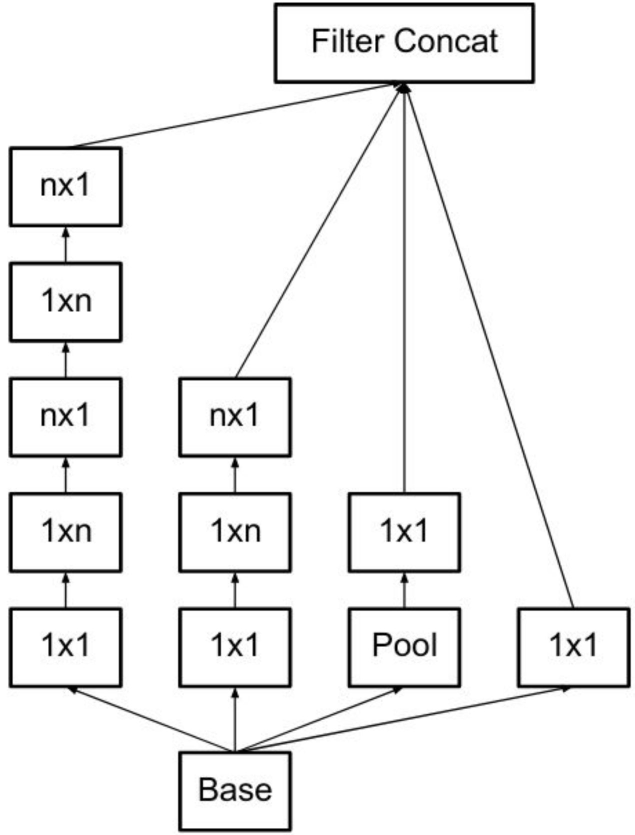
Analytics Vidhya

Module 1



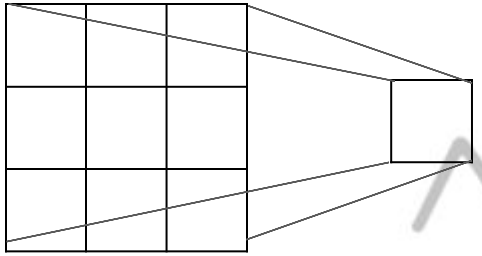
Analytics Vidhya

High Computational Cost
Solution: Factorized Convolution

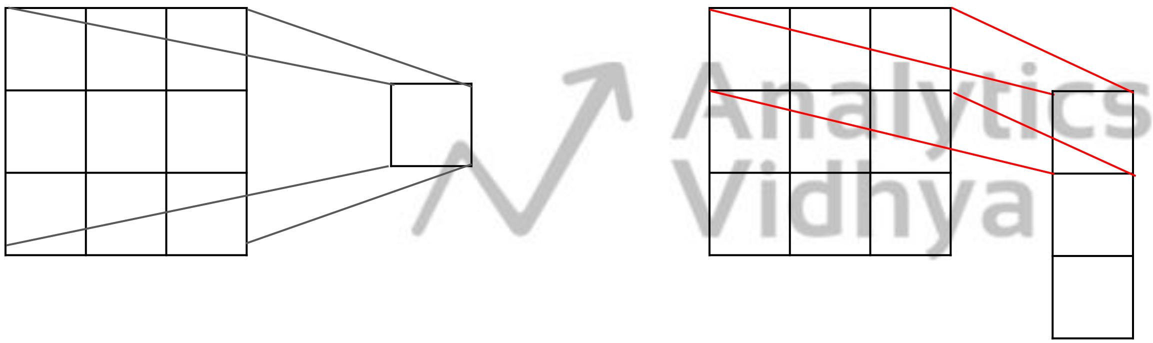


Analytics Vidhya

High Computational Cost
Solution: Factorized Convolution

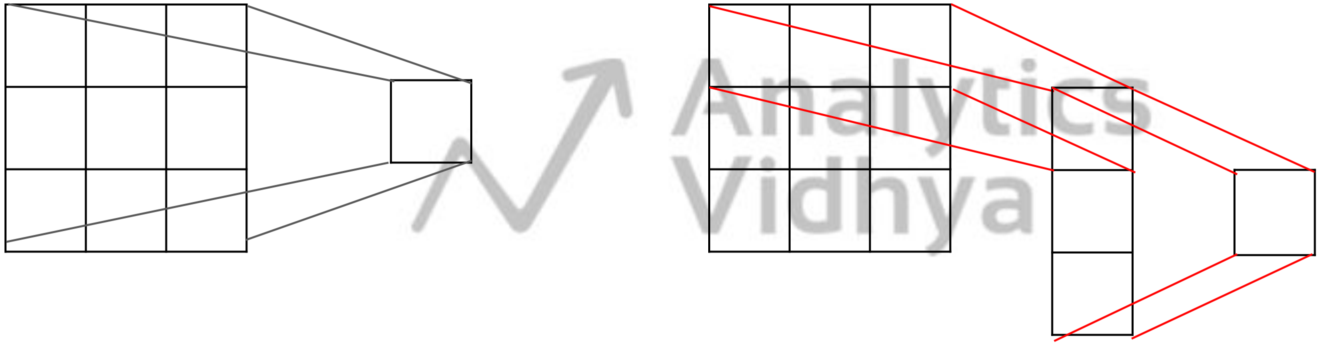


High Computational Cost
Solution: Factorized Convolution

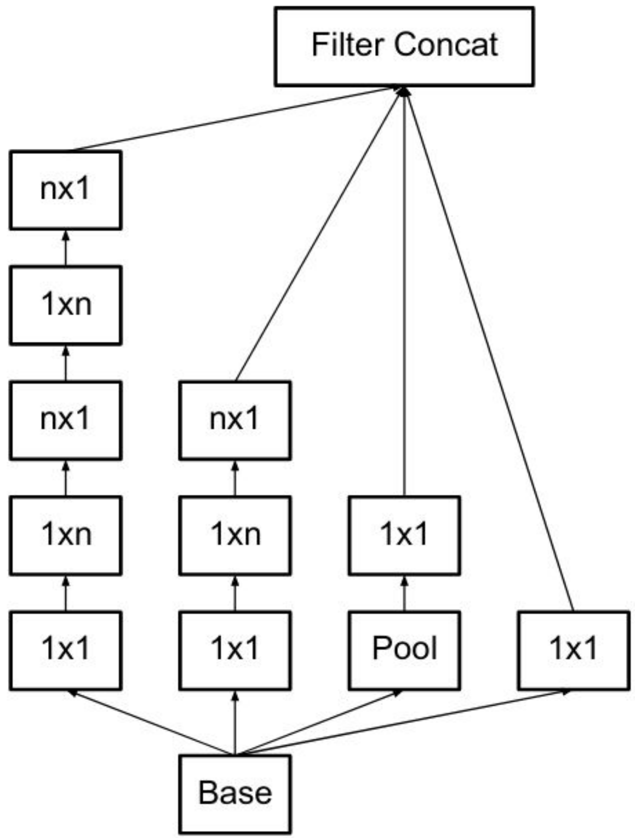


Analytics Vidhya

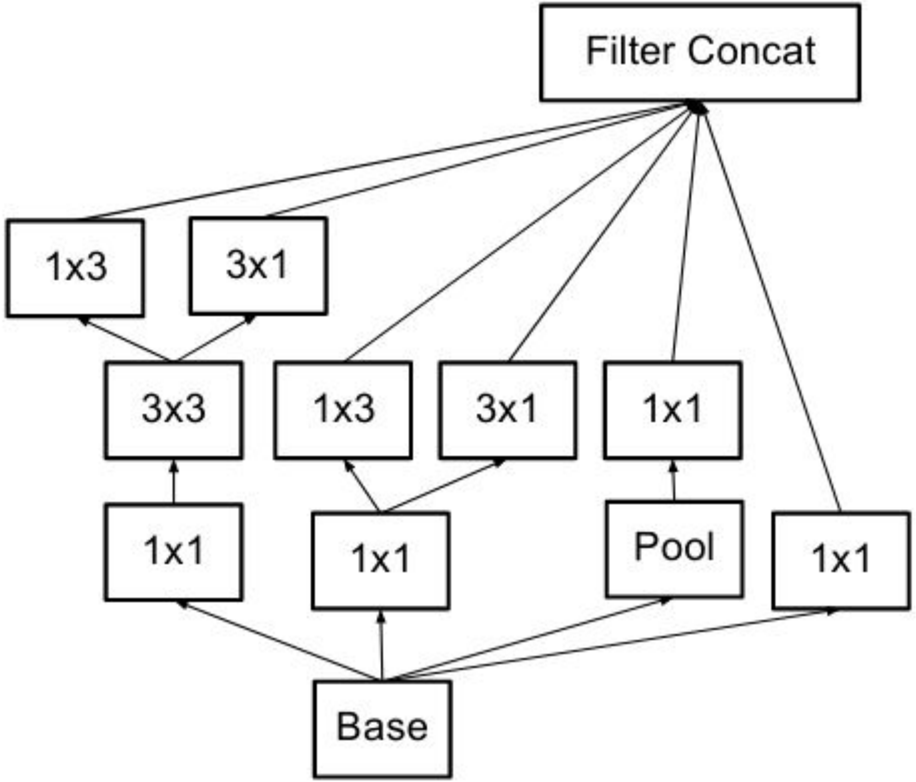
High Computational Cost
Solution: Factorized Convolution



Module 2

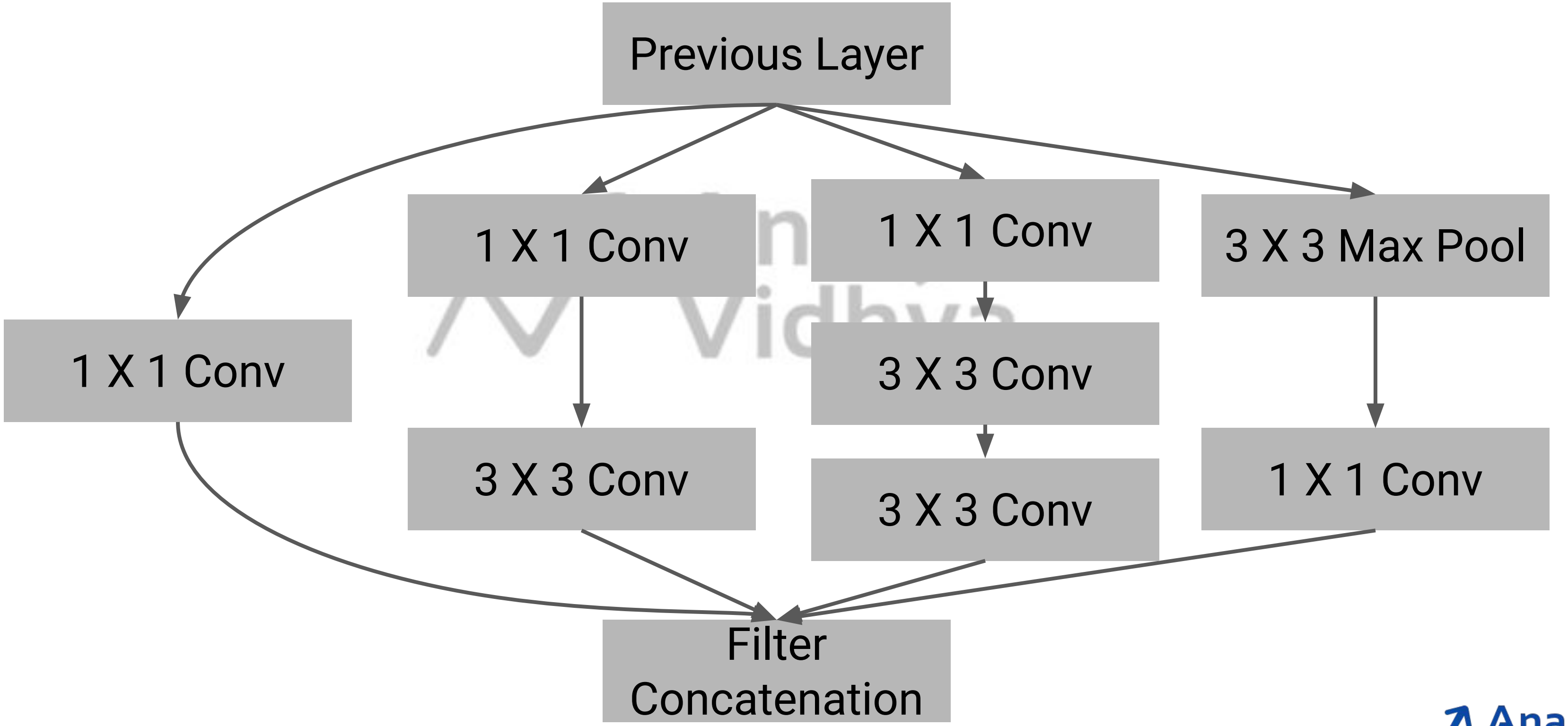

Filter Concat
nx1
1xn
nx1
1xn
1x1
nx1
1xn
1x1
1x1
Pool
1x1
Base

Wider filters

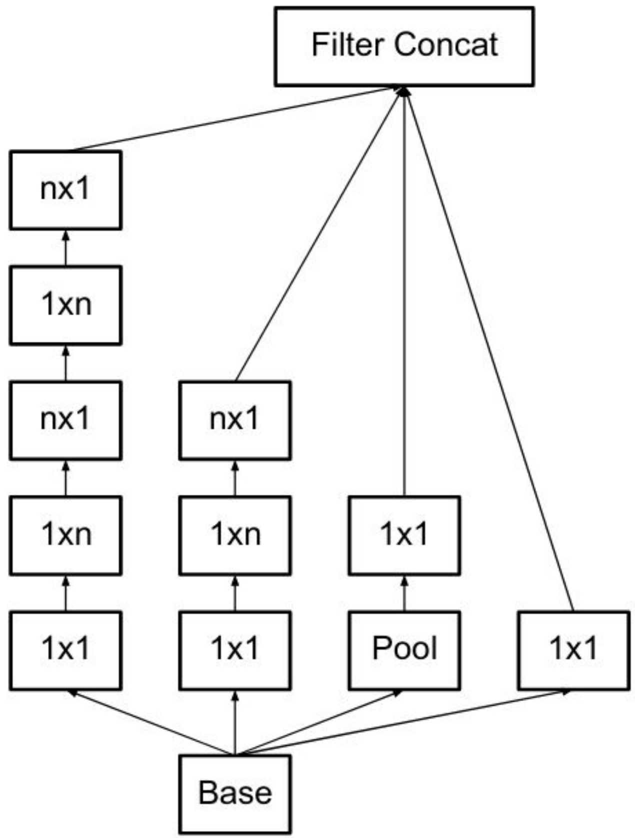


Analytics Vidhya

Module 1

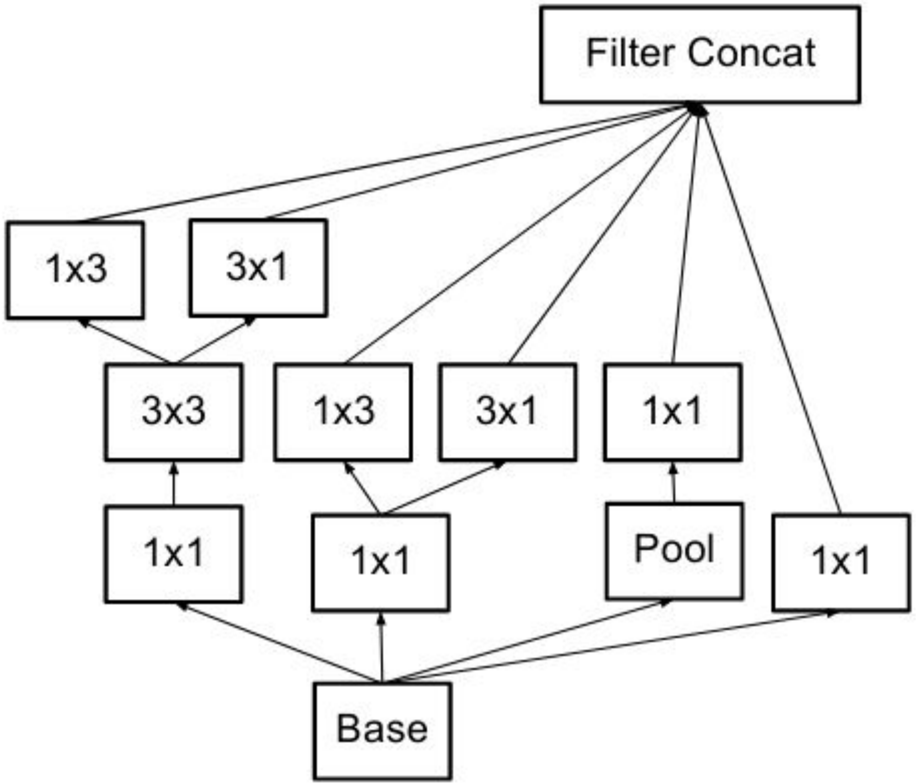

Previous Layer
1 X 1 Conv
1 X 1 Conv
1 X 1 Conv
3 X 3 Max Pool
3 X 3 Conv
3 X 3 Conv
3 X 3 Conv
1 X 1 Conv
Filter Concatenation

Module 2



Analytics Vidhya

Module 3



Inception V2

- Proposed in 2015

Rethinking the Inception Architecture for Computer Vision

Christian Szegedy
Google Inc.
szegedy@google.com

Vincent Vanhoucke
vanhoucke@google.com

Sergey Ioffe
sioffe@google.com

Jonathon Shlens
shlens@google.com

Zbigniew Wojna
University College London
zbigniewwojna@gmail.com

Abstract

Convolutional networks are at the core of most state-of-the-art computer vision solutions for a wide variety of tasks. Since 2014 very deep convolutional networks started to become mainstream, yielding substantial gains in various benchmarks. Although increased model size and computational cost tend to translate to immediate quality gains for most tasks (as long as enough labeled data is provided for training), computational efficiency and low parameter count are still enabling factors for various use cases such as mobile vision and big-data scenarios. Here we are explor-

larly high performance in the 2014 ILSVRC [16] classification challenge. One interesting observation was that gains in the classification performance tend to transfer to significant quality gains in a wide variety of application domains. This means that architectural improvements in deep convolutional architecture can be utilized for improving performance for most other computer vision tasks that are increasingly reliant on high quality, learned visual features. Also, improvements in the network quality resulted in new application domains for convolutional networks in cases where AlexNet features could not compete with hand engineered, crafted solutions, e.g. proposal generation in detection[4].

Analytics Vidhya

Inception V2

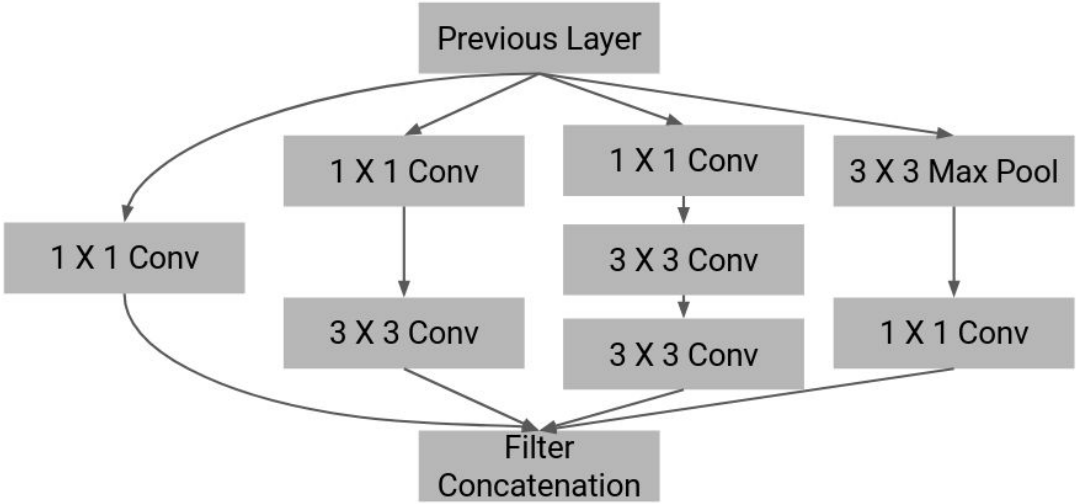
- Proposed in 2015
- It has 42 layers

Analytics Vidhya

Analytics Vidhya

Inception V2

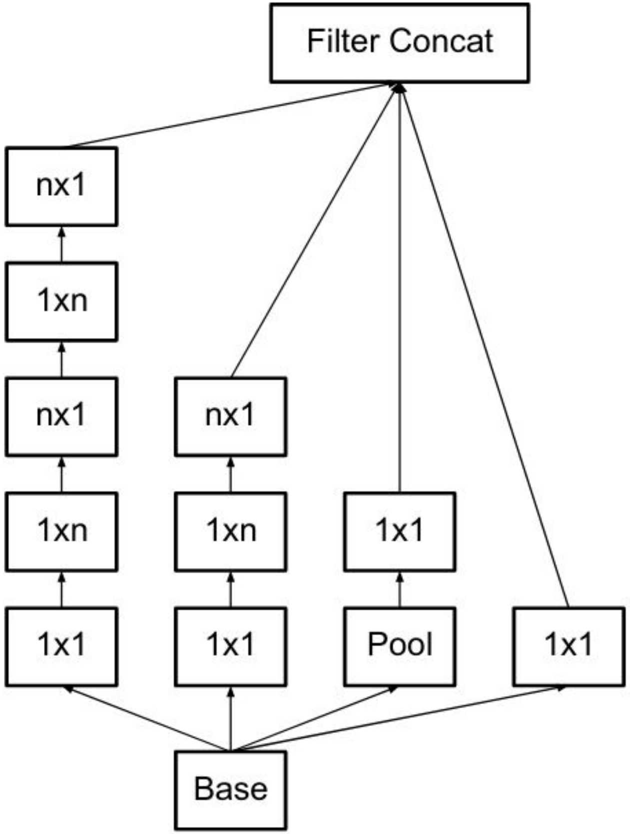
- Proposed in 2015
- It has 42 layers
- Architectural details:
  - 3 Inception module 1



Analytics Vidhya

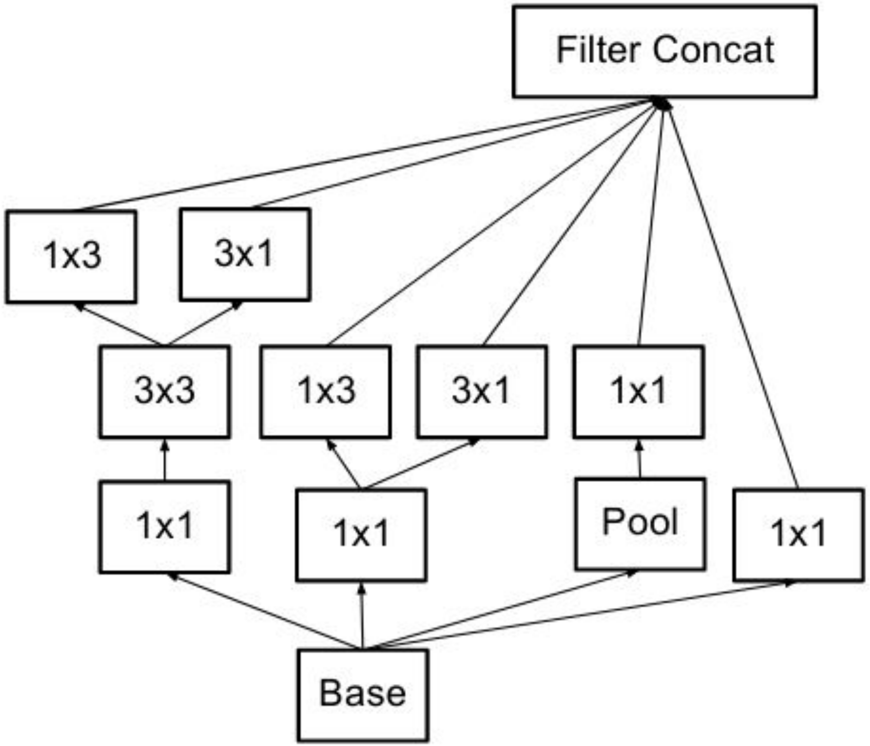
Inception V2

- Proposed in 2015
- It has 42 layers
- Architectural details:
    - 3 Inception module 1
    - 5 Inception module 2



Inception V2

- Proposed in 2015
- It has 42 layers
- Architectural details:
  - 3 Inception module 1
  - 5 Inception module 2
  - 2 Inception module 3



Analytics Vidhya

Inception V2

- Proposed in 2015
- It has 42 layers
- Architectural details:
    - 3 Inception module 1
    - 5 Inception module 2
    - 2 Inception module 3
    - Global average pooling

Analytics Vidhya

Analytics Vidhya

Inception V2

- Proposed in 2015
- It has 42 layers
- Architectural details:
    - 3 Inception module 1
    - 5 Inception module 2
    - 2 Inception module 3
    - Global average pooling
    - 2 Fully connected layers

Analytics Vidhya

Inception V2

- Proposed in 2015
- It has 42 layers
- Architectural details:
    - 3 Inception module 1
    - 5 Inception module 2
    - 2 Inception module 3
    - Global average pooling
    - 2 Fully connected layers
- Trained on ImageNet dataset

Analytics Vidhya

Architecture: Inception V2

type	patch size/stride or remarks	input size
conv	$3\times3/2$	$299\times299\times3$
conv	$3\times3/1$	$149\times149\times32$
conv padded	$3\times3/1$	$147\times147\times32$
pool	$3\times3/2$	$147\times147\times64$
conv	$3\times3/1$	$73\times73\times64$
conv	$3\times3/2$	$71\times71\times80$
conv	$3\times3/1$	$35\times35\times192$
$3\times$Inception	As in figure 5	$35\times35\times288$
$5\times$Inception	As in figure 6	$17\times17\times768$
$2\times$Inception	As in figure 7	$8\times8\times1280$
pool	8×8	$8\times8\times2048$
linear	logits	$1\times1\times2048$
softmax	classifier	$1\times1\times1000$

Analytics Vidhya

Inception V3

Analytics Vidhya

Analytics Vidhya

Inception V3

- RMSProp optimizer

Analytics Vidhya

Analytics Vidhya

Inception V3

- RMSProp optimizer
- Only 1 auxiliary classifier is used

Analytics Vidhya

Analytics Vidhya

Inception V3

- RMSProp optimizer
- Only 1 auxiliary classifier is used
- Added Batch Normalization in the auxiliary classifier

Analytics Vidhya

Comparing versions of Inception

Model	Top-5 Error

Analytics Vidhya

Analytics Vidhya

Comparing versions of Inception

Model	Top-5 Error
GoogLeNet / Inception V1	7.89%

Analytics Vidhya

Comparing versions of Inception

Model	Top-5 Error
GoogLeNet / Inception V1	7.89%
Inception V2	5.82%

Analytics Vidhya

Comparing versions of Inception

Model	Top-5 Error
GoogLeNet / Inception V1	7.89%
Inception V2	5.82%
Inception V3	4.2%

Analytics Vidhya

Analytics Vidhya

Thank You