Question 1: Autoencoder

Autoencoder

Consider grayscale image data with 128×128 pixels. We consider two autoencoder architectures with Rectified Linear Units as nonlinearities:

- **Dense**:
  - Input layer
  - Dense hidden layer with 256 neurons
  - Dense output layer.
- **Convolutional**:
  - Input layer
  - Strided convolutional layer with 64 filters, stride 8, "valid" padding (no zeros added) and a filter size of 8×8.
  - Max-Pooling across the 64 channels.
  - Strided transposed convolutional layer with 64 filters, stride 8 that recover(s) the input (image) dimensions as output.
  - Sum-Pooling across the 64 channels.

Ignoring the bias parameters, how many parameters do we have?

Dense network:
Input Layer: 128x128 $=$ 16348 neurons
Hidden Layer: 256 neurons
Output Layer: 128x128 $=$ 16348 neurons
$\Rightarrow 16348 \cdot 256 + 256 \cdot 16348 = 8388608$ parameters

Question 1: Autoencoder

Autoencoder

Consider grayscale image data with 128×128 pixels. We consider two autoencoder architectures with Rectified Linear Units as nonlinearities:

- **Dense**:
  - Input layer
  - Dense hidden layer with 256 neurons
  - Dense output layer.
- **Convolutional**:
  - Input layer
  - Strided convolutional layer with 64 filters, stride 8, "valid" padding (no zeros added) and a filter size of 8×8.
  - Max-Pooling across the 64 channels.
  - Strided transposed convolutional layer with 64 filters, stride 8 that recover(s) the input (image) dimensions as output.
  - Sum-Pooling across the 64 channels.

Ignoring the bias parameters, how many parameters do we have?

Convolutional network:

Input Layer: no parameters, dimension: 128x128x1

Convolutional Layer: $64 \cdot 8x8 = 4096$ parameters, dimension: 16x16x64

Max-Pooling Layer: no parameters, dimension: 16x16x1

Transposed Convolutional Layer: $64 \cdot 8x8 = 4096$ parameters, dimension: 128x128x64

Sum-Pooling Layer: no parameters, dimension: 128x128x1

$\Rightarrow 4096 + 4096 = 8192$ parameters

Question 2: PCA

Given a real-valued dataset $\mathbf{X}$ and the covariance matrix $\mathbf{C} = \mathbf{X}^T\mathbf{X}$.

Which of the following statements is not true?

A. The PCA eigenvalues can be imaginary

Not True: Let λ be an eigenvalue of C with eigenvector v. Then

$$\lambda||v||^2 = v^T\lambda v = v^T Cv = v^T X^T Xv = ||Xv||^2,$$

so $\lambda = \frac{||Xv||^2}{||v||^2} \in \mathbb{R}$.

Question 2: PCA

Given a real-valued dataset $\mathbf{X}$ and the covariance matrix $\mathbf{C} = \mathbf{X}^T\mathbf{X}$.

Which of the following statements is not true?

A. The PCA eigenvalues can be imaginary

Not True: Let λ be an eigenvalue of C with eigenvector v. Then

$$\lambda||v||^2 = v^T\lambda v = v^T C v = v^T X^T X v = ||Xv||^2,$$

so $\lambda = \frac{||Xv||^2}{||v||^2} \in \mathbb{R}$.

B. $\mathbf{C} = \mathbf{C}^T$

True: $C^T = (X^T X)^T = X^T(X^T)^T = X^T X = C$.

Question 2: PCA

Given a real-valued dataset $\mathbf{X}$ and the covariance matrix $\mathbf{C} = \mathbf{X}^T\mathbf{X}$.

Which of the following statements is not true?

✔ ◉ **A.** The PCA eigenvalues can be imaginary

Not True: Let λ be an eigenvalue of C with eigenvector v. Then

$$\lambda||v||^2 = v^T \lambda v = v^T C v = v^T X^T X v = ||Xv||^2,$$

so $\lambda = \frac{||Xv||^2}{||v||^2} \in \mathbb{R}$.

○ **B.** $\mathbf{C} = \mathbf{C}^T$

True: $C^T = (X^T X)^T = X^T (X^T)^T = X^T X = C$.

○ **C.** The PCA eigenvalues are always non-negative

True: As seen in A, we know that for any eigenvalue λ of C with eigenvector v, it holds that $\lambda = \frac{||Xv||^2}{||v||^2} \geq 0$.

Question 3: Multiple Applications of PCA

Consider a dataset $\mathbf{X} \in \mathbb{R}^{T \times n}$. We perform a PCA on the dataset $\mathbf{X}$ by computing the covariance matrix

$$\mathbf{C}^{(1)} = \frac{\mathbf{X}^T\mathbf{X}}{T-1}$$

and solving the eigenvalue problem

$$\mathbf{C}^{(1)}\mathbf{w}_i^{(1)} = \left(\sigma_i^{(1)}\right)^2 \mathbf{w}_i^{(1)}.$$

Projecting onto the $m < n$ principal components $\mathbf{W}_m^{(1)} = \left[\mathbf{w}_1^{(1)}, \ldots, \mathbf{w}_m^{(1)}\right]$ with the largest eigenvalues yields

$$\mathbf{Y} = \mathbf{X}\mathbf{W}_m^{(1)}.$$

We perform a PCA a second time by computing

$$\mathbf{C}^{(2)} = \frac{\mathbf{Y}^T\mathbf{Y}}{T-1}$$

$$\mathbf{C}^{(2)}\mathbf{w}_i^{(2)} = \left(\sigma_i^{(2)}\right)^2 \mathbf{w}_i^{(2)}.$$

Projecting onto the single principal component $\mathbf{w}_1^{(2)}$ with the largest eigenvalue $\left(\sigma_1^{(2)}\right)^2$ results in

$$\mathbf{z}_1 = \mathbf{Y}\mathbf{w}_1^{(2)}$$

Which of the following is true in general?

Question 3: Multiple Applications of PCA

○ **A.** $\left(\sigma_1^{(1)}\right)^2 < \left(\sigma_1^{(2)}\right)^2$

Not True: By construction, $C^{(2)} = Y^T Y$ is a diagonal matrix with the diagonal entries $(\sigma_i^{(1)})^2$. The eigenvectors are the unit vectors, and the new eigenvalues are the same as before.

○ **B.** $\mathbf{w}_1^{(1)} = \mathbf{w}_2^{(2)}$

Not True: In general, the eigenvectors $w_1^{(1)}$ are not unit vectors, the second eigenvectors however are unit vectors.

Question 3: Multiple Applications of PCA

✔ ◉ **C.** $\mathbf{Y}_{*,1} = \mathbf{z}_1$

True: The vector $w_1^{(2)}$ is a unit vector corresponding to the largest eigenvalue of $Y^T Y$, which is by construction $(\sigma_1^{(1)})^2$. So $w_1^{(2)} = (1, 0, 0, \ldots)^T$. This means that $z_1 = Yw_1^{(2)}$ is the first column of Y.

○ **D.** $\left\| \mathbf{w}_1^{(1)} - \mathbf{w}_1^{(2)} \right\| = 1$

Not True: We do know that $||w_1^{(1)}|| = ||w_1^{(2)}|| = 1$, but since $w_1^{(1)}$ may be rather arbitrary depending on the data, we do not know anything about the difference in general.

Question 4: Data analysis with PCA

In this exercise you are given a data set $\mathbf{X} \in \mathbb{R}^{n\times 4}$ (for $n \to \infty$) which has been i.i.d. sampled according to some source distribution $P(x)$. We assume that P is more or less behaving like a multivariate normal distribution, with zero mean and unknown covariance.

For this data, you compute the covariance matrix:

$$\mathbf{C} = \mathbf{X}\mathbf{X}^T = \begin{bmatrix} 9 & -6 & 3 & -3 \\ -6 & 4 & -2 & 2 \\ 3 & -2 & 1 & -1 \\ -3 & 2 & -1 & 1 \end{bmatrix}.$$

Given a new data point $\mathbf{x} = (x_0, x_1, x_2, x_3)^T \sim P(\mathbf{x})$ sampled from the same source, you observe that $x_0 = 15$.

Hint 1: You do not have to calculate anything by hand, if you study $\mathbf{C}$ closely.

Hint 2: If you still feel lost, you might want to look at the *linalg* module of *numpy*.

Which of the following statements are likely to be true?

- [ ] **A.** $x_1 \approx -10$ **and** $x_3 \cdot x_2 > 0$.
- [x] **B.** $x_1 \approx -10$ **and** $x_3 \cdot x_2 < 0$.
- [ ] **C.** $x_1 \approx 6$ **and** $x_3 \cdot x_2 > 0$.
- [ ] **D.** $x_1 \approx -6$ **and** $x_3 \cdot x_2 > 0$.
- [x] **E.** $x_1 < 0$ **and** $x_3 \approx x_2 \approx 0$.

Looking at the covariance matrix, the correlation between x_0 and x_1 is -6. On the other hand, the variance of x_0 is 9 and of x_1 is 4. This means that the values of x_0 and x_1 are likely to have opposite signs. Also, since $\frac{9}{-6} = \frac{-6}{4} = -\frac{3}{2}$, we can deduce that it is likely to have $x_1 \approx -10$. Also, the correlation of x_3 and x_2 is -1, i. e. negative, so opposite signs are again likely, which implies $x_3 \cdot x_2 < 0$.

Question 4: Data analysis with PCA

In this exercise you are given a data set $\mathbf{X} \in \mathbb{R}^{n\times 4}$ (for $n \to \infty$) which has been i.i.d. sampled according to some source distribution $P(x)$. We assume that P is more or less behaving like a multivariate normal distribution, with zero mean and unknown covariance.

For this data, you compute the covariance matrix:

$$\mathbf{C} = \mathbf{X}\mathbf{X}^T = \begin{bmatrix} 9 & -6 & 3 & -3 \\ -6 & 4 & -2 & 2 \\ 3 & -2 & 1 & -1 \\ -3 & 2 & -1 & 1 \end{bmatrix}.$$

Given a new data point $\mathbf{x} = (x_0, x_1, x_2, x_3)^T \sim P(\mathbf{x})$ sampled from the same source, you observe that $x_0 = 15$.

Hint 1: You do not have to calculate anything by hand, if you study $\mathbf{C}$ closely.

Hint 2: If you still feel lost, you might want to look at the *linalg* module of *numpy*.

Which of the following statements are likely to be true?

- [ ] **F. All eigenvalues of C are non-zero.**
- [x] **G. There is exactly one axis of variance describing the data distribution, the rest is negligible noise.** ✔
- [ ] **H. There are exactly two axes of variance describing the data distribution, the rest is negligible noise.**
- [ ] **I. There are exactly three axes of variance describing the data distribution, the rest is negligible noise.**
- [ ] **J. All four axes of variance are necessary to describe the data distribution.**

From the matrix, one can see that all rows are linearly dependent. Each is a scalar multiple of the last row, the scalars being $-3, 2, -1, 1$. Since the matrix doesn't have full rank, the matrix has eigenvalue 0. Also, since the rank of C is 1, there is exactly one axis of variance describing the data distribution.

Question 5: Properties of orthogonal auto-encoders

In this exercise we study auto-encoders. We provide the following definitions and facts as they will be useful to solve the question:

1. A function $f: \mathbb{R}^n \to \mathbb{R}^n$ is called L-Lipschitz continuous under the Euclidean norm $\|\cdot\|_2$, iff for all $x, y \in \mathbb{R}^n$, we have $\|f(x) - f(y)\|_2 \leq L\|x - y\|_2$.
2. A matrix $A \in \mathbb{R}^{m \times n}$ for $m < n$ is called semi-orthogonal iff $AA^T = I_m$, where I_m denotes the identity matrix on $\mathbb{R}^m$.
3. The spectral norm of a matrix $A \in \mathbb{R}^{m \times n}$, denoted as $\|A\|_2$ is given by $\|A\|_2 = \sqrt{\lambda_{\max, A^T A}}$ where $\lambda_{\max, A^T A}$ is the biggest Eigenvalue of $A^T A$.
4. For any matrix $A \in \mathbb{R}^{m \times n}$ and any vector $x \in \mathbb{R}^n$ the so-called operator inequality holds

$$\|Ax\|_2 \leq \|A\|_2 \|x\|_2$$

Now we study an auto-encoder that encodes vectors to a lower dimensional code given the following form:

- let $n > m > k > 0$ be integers.
- let $\rho: \mathbb{R} \to \mathbb{R}$ be the ReLU activation function $\rho(x) = \max(0, x)$.
- let $U_1 \in \mathbb{R}^{m \times n}, U_2 \in \mathbb{R}^{k \times m}, V_1 \in \mathbb{R}^{m \times n}, V_2 \in \mathbb{R}^{k \times m}$ be semi-orthogonal matrices.
- let $a_1 \in \mathbb{R}^m, a_2 \in \mathbb{R}^k, b_1 \in \mathbb{R}^n, b_2 \in \mathbb{R}^m$ be bias vectors.

We then define the encoder map to be

$$z = e(x) = U_2 \rho(U_1 x + a_1) + a_2$$

and the decoder map to be

$$x = d(z) = V_1^T \rho(V_2^T z + b_2) + b_1.$$

Now assume you are given x, y, z such that $\|x - y\|_2 < 1$ and $\|x - z\|_2 < 1$.

Given this information, which one of the following statements is correct (in general)?

Question 5: Properties of orthogonal auto-encoders

✔ ◉ **A.** The concatenation of d and e given by $(d \circ e)(x) := d(e(x))$ (also called reconstruction) is π-Lipschitz continuous.

True: Considering the ReLU activation function ρ and the difference $|\rho(x) - \rho(y)|$, we have the following three cases:

- Case 1: $x, y \leq 0$: $|\rho(x) - \rho(y)| = 0 \leq |x - y|$.
- Case 2: $x, y > 0$: $|\rho(x) - \rho(y)| = |x - y|$.
- Case 3: $x > 0, y \leq 0$: $|\rho(x) - \rho(y)| = x \leq x - y = |x - y|$.
- Case 4: $x \leq 0, y > 0$: See case 3.

So ρ is 1-Lipschitz continuous.
Additionally, any map of the form $f(x) = x + b$ is trivially 1-Lipschitz continuous.

Question 5: Properties of orthogonal auto-encoders

In general, for any linear map A we have that

$$||Ax - Ay||_2 \leq ||A||_2 \cdot ||x - y||_2,$$

i. e. they are $||A||_2$-Lipschitz-continuous.
Now lets consider a semi-orthogonal matrix $U \in \mathbb{R}^{m \times n}$, i. e. $UU^T = I_m$.
What can we say about $||U||_2$?

Question 5: Properties of orthogonal auto-encoders

In general, for any linear map A we have that

$$||Ax - Ay||_2 \leq ||A||_2 \cdot ||x - y||_2,$$

i. e. they are $||A||_2$-Lipschitz-continuous.
Now lets consider a semi-orthogonal matrix $U \in \mathbb{R}^{m \times n}$, i. e. $UU^T = I_m$.
What can we say about $||U||_2$?
Well, let λ be an eigenvalue of $U^T U$ with eigenvector v. Then we can see that:

$$Uv = I_m Uv = UU^T Uv = \lambda Uv \Rightarrow Uv = 0 \text{ or } \lambda = 1.$$

Since U must have rank m, we immediately know that $||U||_2 = 1$.

Question 5: Properties of orthogonal auto-encoders

In general, for any linear map A we have that

$$||Ax - Ay||_2 \leq ||A||_2 \cdot ||x - y||_2,$$

i. e. they are $||A||_2$-Lipschitz-continuous.
Now lets consider a semi-orthogonal matrix $U \in \mathbb{R}^{m \times n}$, i. e. $UU^T = I_m$.
What can we say about $||U||_2$?
Well, let λ be an eigenvalue of $U^T U$ with eigenvector v. Then we can see that:

$$Uv = I_m Uv = UU^T Uv = \lambda Uv \Rightarrow Uv = 0 \text{ or } \lambda = 1.$$

Since U must have rank m, we immediately know that $||U||_2 = 1$.
On the other hand, what is $||U^T||_2$? Well, this one is more obvious:

$$||U^T||_2 = \sqrt{\lambda_{UU^T,\max}} = \sqrt{\lambda_{I_m,\max}} = 1.$$

So in both cases, the linear map is 1-Lipschitz continuous.

Question 5: Properties of orthogonal auto-encoders

It is easy to see that if f is a-Lipschitz continuous, and g is b-Lipschitz continuous, then $g \circ f$ is $a \cdot b$-Lipschitz continuous:

$$||g(f(x)) - g(f(y))||_2 \leq b \cdot ||f(x) - f(y)||_2 \leq a \cdot b \cdot ||x - y||_2.$$

Question 5: Properties of orthogonal auto-encoders

It is easy to see that if f is a-Lipschitz continuous, and g is b-Lipschitz continuous, then $g \circ f$ is $a \cdot b$-Lipschitz continuous:

$$||g(f(x)) - g(f(y))||_2 \leq b \cdot ||f(x) - f(y)||_2 \leq a \cdot b \cdot ||x - y||_2.$$

So back to our objective. Our Encoder/Decoder functions are just concatenations of:

- Linear maps with semi-orthogonal matrices (or their transposed version),
- Maps that are adding bias vectors,
- and the ReLU activation function.

All of these are 1-Lipschitz continuous, so their concatenation is also 1-Lipschitz continuous, and especially π-Lipschitz continuous.

Question 5: Properties of orthogonal auto-encoders

○ **B.** $\|d(e(y)) - d(e(z))\|_2 > \pi$.

Not True: Counterexample: y and z were chosen arbitrarily, only constrained by $||y - z||_2 < 1$. But if we chose $y = z$, the constraint is fulfilled, and $||d(e(y)) - d(e(z))||_2 = 0 \leq \pi$.

Question 5: Properties of orthogonal auto-encoders

○ **B.** $\|d(e(y)) - d(e(z))\|_2 > \pi$.

Not True: Counterexample: y and z were chosen arbitrarily, only constrained by $||y - z||_2 < 1$. But if we chose $y = z$, the constraint is fulfilled, and $||d(e(y)) - d(e(z))||_2 = 0 \leq \pi$.

○ **C.** The encoder map e projects x onto its first k principal components.

Not True: In general, this is not true. The matrices U_1 and U_2 are chosen arbitrarily, and do not have to match the data set in any way, and do not have to fulfill the claim.

Question 5: Properties of orthogonal auto-encoders

B. $\|d(e(y)) - d(e(z))\|_2 > \pi$.

Not True: Counterexample: y and z were chosen arbitrarily, only constrained by $||y - z||_2 < 1$. But if we chose $y = z$, the constraint is fulfilled, and $||d(e(y)) - d(e(z))||_2 = 0 \leq \pi$.

C. The encoder map e projects x onto its first k principal components.

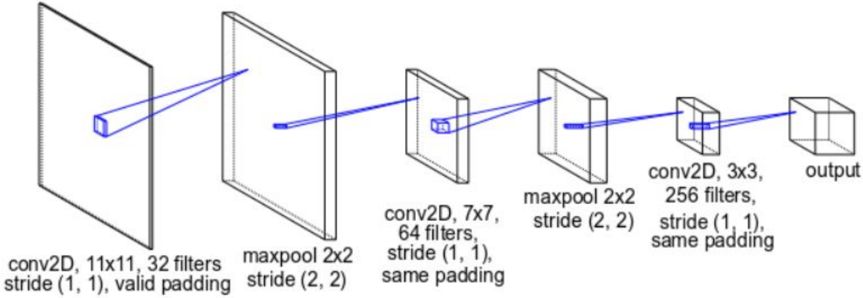
Not True: In general, this is not true. The matrices U_1 and U_2 are chosen arbitrarily, and do not have to match the data set in any way, and do not have to fulfill the claim.

D. There is a combination of U_1, U_2, V_1, V_2 such that $(d \circ e)(x) = x$ for all $x \in \mathbb{R}^n$.

Not True: After going through the decoder, we land in the latent space of dimension m. Since $m < n$, it is impossible to recover the input from the value in the latent space without loss.

Question 6: Convolutional Neural Network

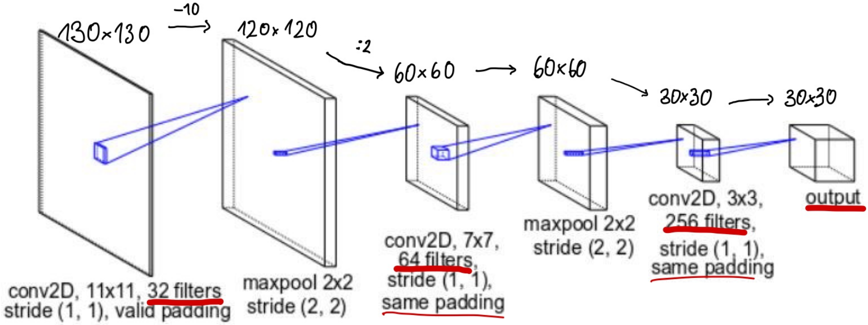
Given a dataset with 130 × 130 px images and 3 channels, consider the following convolutional neural network:



What is the output dimension?

Question 6: Convolutional Neural Network

Given a dataset with 130 × 130 px images and 3 channels, consider the following convolutional neural network:



What is the output dimension? $25 \times 25 \times 256$

Question 7: Kernels

We have a Gaussian function in two dimensions given by

$$G(x,y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}}$$

One can use such a Gaussian to parameterize a convolution kernel $K \in \mathbb{R}^{n\times n}$ by, e.g., spanning an odd-sized equidistant grid centered around $(0,0)^T$, evaluating G on the grid points, and normalizing the resulting matrix with respect to the sum of its values.

Can K be written as an outer product $K = uv^T$, where $u, v \in \mathbb{R}^n$?

Consider the vectors u, v with

$$u_i = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{x_i^2}{2\sigma^2}}, \quad v_j = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{y_j^2}{2\sigma^2}}$$

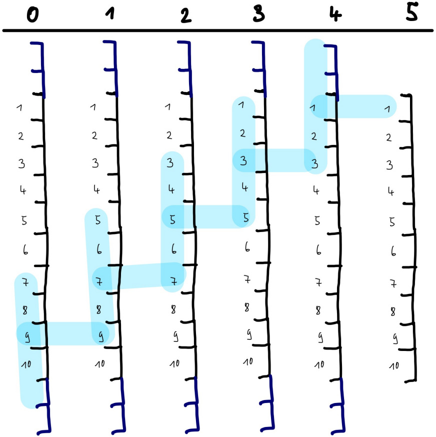
where x_i, y_j are the grid points around $(0,0)^T$. Then we can see that

$$G(x_i, y_j) = \frac{1}{2\pi\sigma^2} e^{-\frac{x_i^2+y_j^2}{2\sigma^2}} = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{x_i^2}{2\sigma^2}} \cdot \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{y_j^2}{2\sigma^2}} = u_i \cdot v_j = (uv^T)_{ij}$$

In general, this is not the case for arbitrary K, since any K of the form uv^T has rank one, which is not always the case for $K \in \mathbb{R}^{n\times n}$.

Question 8: Sparsity in convolutional neural networks

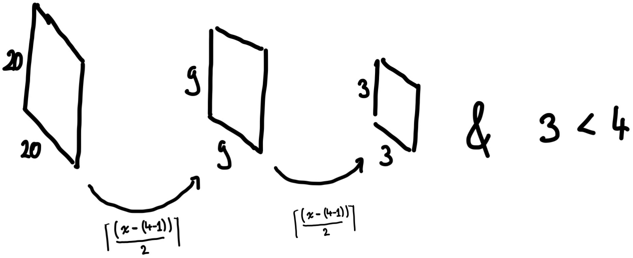
Given images of dimension 8×10 and a network of depth d that contains the same convolutional layer d times with kernel size 5×5 , stride 1, and same padding. At which minimum depth does every feature pixel in the last layer depend on all input pixels?



Question 9: Valid padding

Given a concatenation of d identical convolutional layers with valid padding, kernel size 4×4, stride 2 and input image size 20×20.

At which depth d of the network do the kernel dimensions exceed the dimensions of the layer's input?

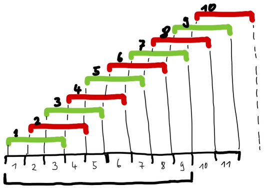


$$\text{"new SL"} = \left\lceil \frac{\text{"old SL"} - (\text{"kernel size"} - 1)}{\text{"stride"}} \right\rceil$$

Question 10: Output dimensions

Consider a convolution kernel of size $n \times n$, a stride s, and an input image of size $w \times h$.

Assuming w, h, and n being divisible by the stride s, what is the output dimension when applying valid padding?



$$\text{"new SL"} = \left\lceil \frac{\text{"old SL"} - (\text{"kernel size"} - 1)}{\text{"stride"}} \right\rceil$$

$$= \frac{\text{"old SL"} - \text{"kernel size"}}{\text{"stride"}} + \underbrace{\left\lceil \frac{1}{\text{"stride"}} \right\rceil}_{=1}$$

✔ A. $[(w-n)/s+1] \times [(h-n)/s+1]$