DUMPSBOSS.

IBM AI Enterprise Workflow V1 Data Science Specialist

IBM C1000-059

Version Demo

Total Demo Questions: 8

Total Premium Questions: 85

Buy Premium PDF

https://dumpsboss.co

support@dumpsboss.co

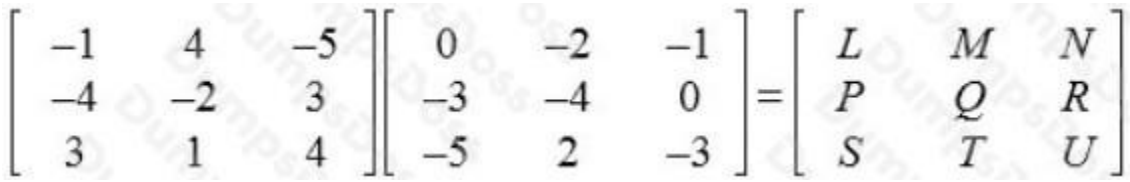
support@dumpsboss.co
dumpsboss.co

Topic Break Down

Topic	No. of Questions
Topic 1, Business Understanding	7
Topic 2, Data Understanding	10
Topic 3, Data Preparation	22
Topic 4, Modeling	20
Topic 5, Evaluation	15
Topic 6, Deployment	9
Topic 7, Mix Questions	2
Total	85

QUESTION NO: 1

Given the following matrix multiplication:



What is the value of P?

A. –9

B. 17

C. 12

D. –7

ANSWER: C

Explanation:

The value of P is **12**. In matrix multiplication, each entry in the product matrix is obtained by taking the dot product of the corresponding row from the matrix on the left and the corresponding column from the matrix on the right. Thus, to find P, pair each value in P's source row with the value in the matching position of P's source column, multiply each pair, and add those products together. Applying that row-by-column calculation to the matrices shown produces a total of 12.

This follows the standard definition of matrix multiplication: if an entry is in row *i* and column *j* of a product, its value is the sum of the products of corresponding entries from row *i* of the first matrix and column *j* of the second matrix. The operation is valid when the first matrix's number of columns matches the second matrix's number of rows. See Math is Fun: Matrix Multiplication and Wolfram MathWorld: Matrix Multiplication for the row-by-column rule and worked examples.

QUESTION NO: 2

What are two hyperparameters used when building a k-means model? (Choose two.)

A. kernel

B. learning rate

C. number of iterations

D. number of clusters

E. number of neighbors

ANSWER: C D

Explanation:

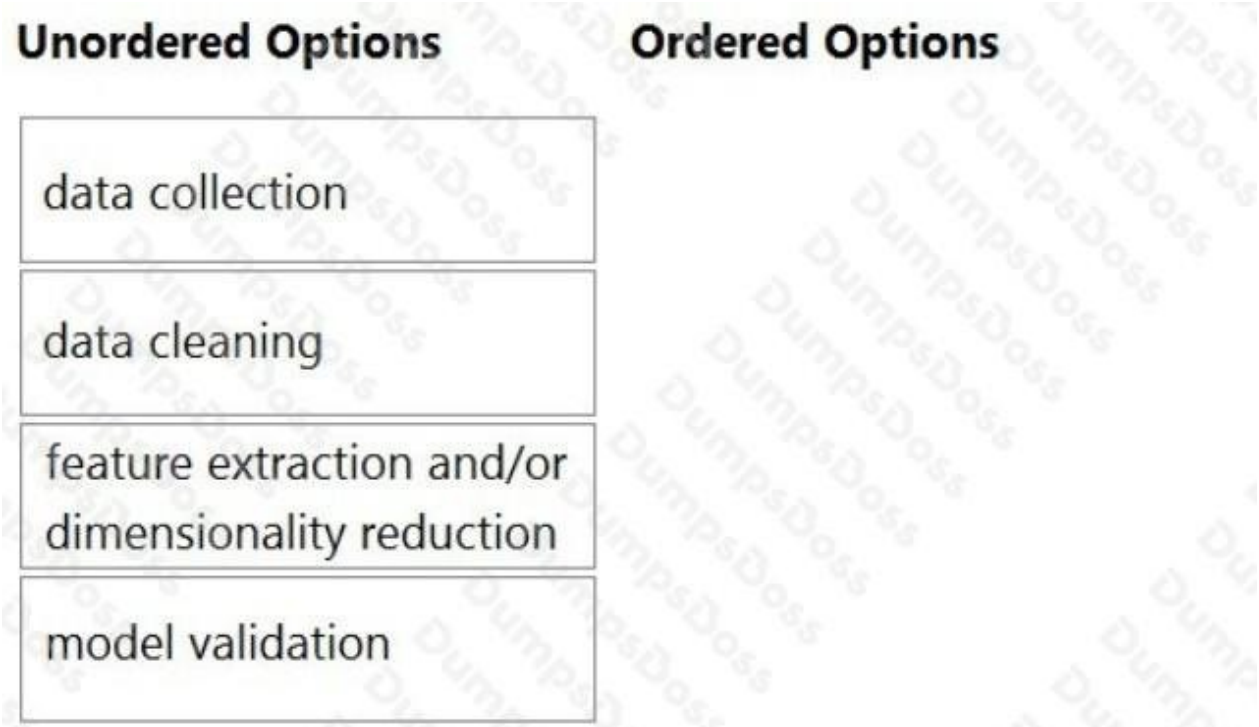
For k-means clustering, **number of clusters** and **number of iterations** are configurable hyperparameters that control the clustering process before model fitting. The number of clusters, commonly denoted by *k*, specifies how many centroid-based groups the algorithm should produce. Selecting this value is fundamental because it defines the intended segmentation granularity of the data. The number of iterations specifies the maximum number of centroid-update cycles allowed while k-means alternates between assigning observations to the nearest centroid and recalculating centroid locations. This limit helps control runtime and provides a stopping safeguard if convergence is slow. In practical implementations, k-means may stop before reaching that maximum when its convergence criterion is met, such as when centroid positions no longer change materially. IBM Watson Studio's machine-learning documentation describes k-means as a clustering method requiring a specified number of clusters, while common k-means implementations expose a maximum-iteration setting to govern optimization. See the IBM k-means documentation and the scikit-learn KMeans API reference.

QUESTION NO: 3 - (DRAG DROP)

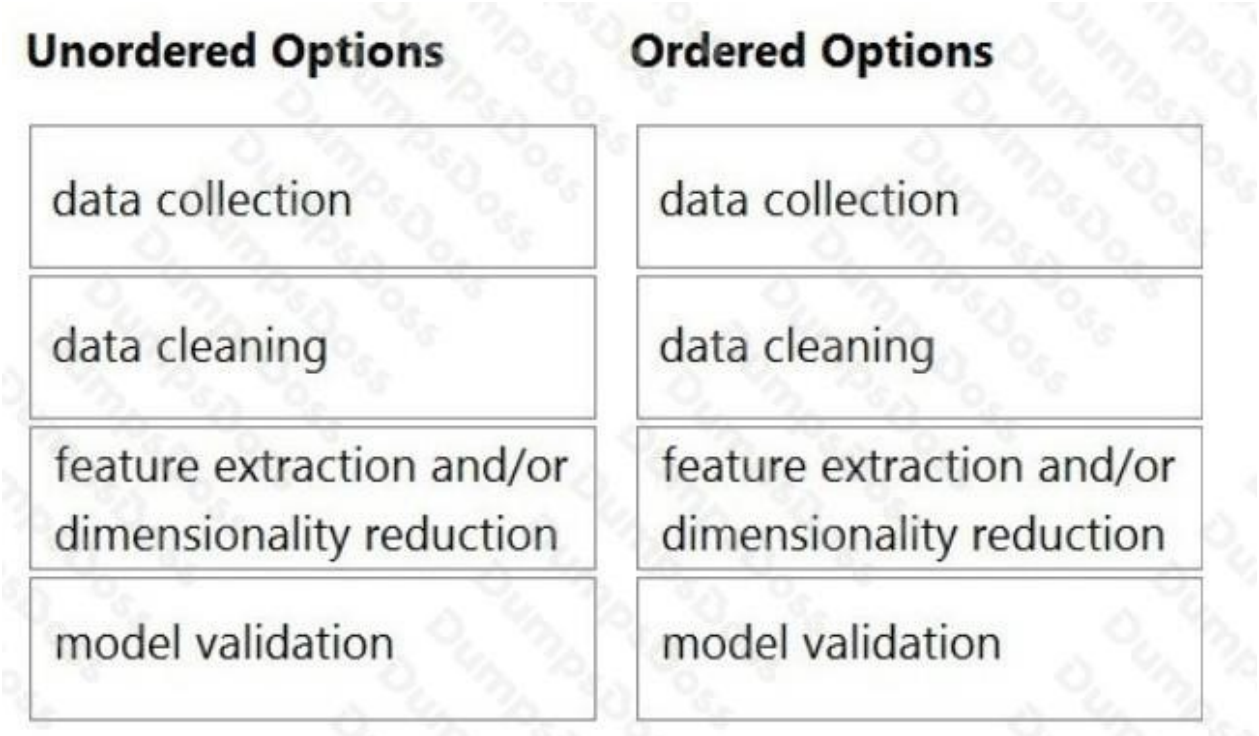
DRAG DROP

What is the best step by step order for machine learning pipeline?

Select and Place:



ANSWER:



Explanation:

A sound machine-learning pipeline begins with **data collection**, because the project first needs the raw observations, records, measurements, or labels from which a model can learn. The quality, coverage, and relevance of the acquired data establish the basis for every later stage. IBM describes data preparation as a process applied to data that has been obtained, so acquisition necessarily precedes preparation. See IBM's overview of data preparation.

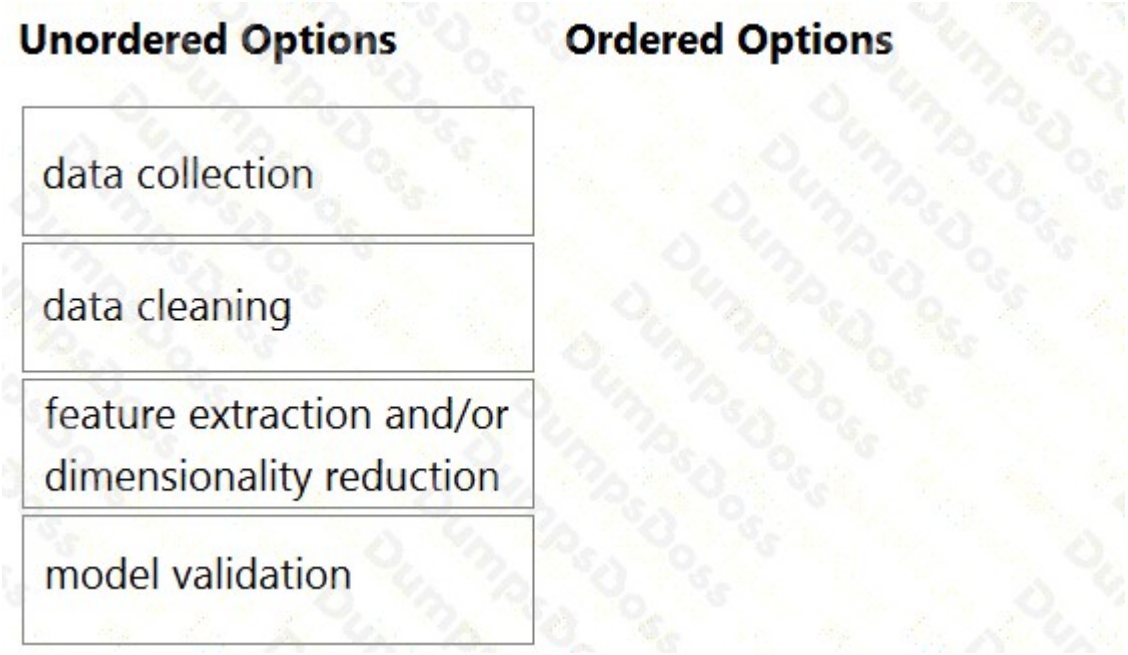
After the data is available, **data cleaning** prepares it for reliable analysis. This stage handles issues such as missing values, invalid records, duplicates, inconsistent formats, and inappropriate data types. Producing a trustworthy and usable dataset at this point helps ensure that subsequent transformations are applied consistently to meaningful input data.

Next, **feature extraction and/or dimensionality reduction** converts prepared data into a representation that is useful for machine learning. Feature extraction derives informative variables from the cleaned source data, while dimensionality reduction can reduce the number of variables while retaining important structure or signal. These transformations are part of preparing the model inputs and should occur before evaluating model performance. IBM's feature engineering discussion explains the importance of creating suitable input features for machine-learning models.

Finally, **model validation** evaluates how well the resulting model generalizes to data not used for fitting. Validation uses an appropriate holdout set or cross-validation process to estimate performance and support model-selection decisions. Performing this activity after the collection, cleaning, and feature-preparation workflow ensures that the evaluated model receives the intended prepared inputs. This progression follows the foundational lifecycle of acquiring data, making it usable, representing it for learning, and then assessing the model.

QUESTION NO: 4 - (SIMULATION)

What is the best step by step order for machine learning pipeline?



ANSWER: See the explanation for the answer

Explanation:

Correct order:

1. Data collection
2. Data cleaning
3. Feature extraction and/or dimensionality reduction
4. Model validation

First obtain the raw data, then correct/remove missing, invalid, duplicate, or inconsistent data. Next create/select useful features and, where appropriate, reduce dimensionality. Finally, validate the resulting model against held-out or cross-validation data to assess its performance.

QUESTION NO: 5

What is a class of machine learning problems where the algorithm builds a mathematical model from a set of data that contains both the inputs and the desired outputs?

A. unsupervised learning

B. mentoring

C. reinforcement learning

D. supervised learning

ANSWER: D

Explanation:

Supervised learning is the class of machine learning described because it trains a model using labeled examples: each training record supplies input features together with the known desired output, often called a target or label. During training, the algorithm learns a mathematical relationship that maps the inputs to that target by minimizing prediction error across the labeled data. Once trained, the resulting model can apply the learned relationship to new records whose outputs are not yet known.

This approach supports both major predictive task types. In classification, the desired output is a discrete category, such as whether a customer is likely to churn. In regression, the desired output is a numeric value, such as an expected sales amount. The essential characteristic in the question is the availability of both inputs and desired outputs in the data used to build the model. IBM describes supervised learning as using labeled datasets to train algorithms to classify data or predict outcomes accurately. See IBM: What is supervised learning? and the IBM watsonx documentation on supervised learning.

QUESTION NO: 6

Which is a preferred approach for simplifying the data transformation steps in machine learning model management and maintenance?

A. Implement data transformation, feature extraction, feature engineering, and imputation algorithms in one single pipeline.

B. Do not apply any data transformation or feature extraction or feature engineering steps.

C. Leverage only deep learning algorithms.

D. Apply a limited number of data transformation steps from a pre-defined catalog of possible operations independent of the machine learning use case.

ANSWER: A

Explanation:

Implement data transformation, feature extraction, feature engineering, and imputation algorithms in one single pipeline. A pipeline makes the sequence of preprocessing operations part of the model's repeatable workflow rather than a collection of separate, manually maintained steps. The same fitted transformations can therefore be applied consistently during training, validation, batch scoring, and deployment. This reduces the risk that production data is prepared differently from training data, which can otherwise cause unreliable predictions and make model maintenance difficult. Keeping the transformations and estimator together also improves reproducibility, versioning, testing, and operational handoff: teams can update or redeploy one governed workflow instead of coordinating multiple independent scripts or processes. IBM's machine-learning lifecycle tooling supports organizing and operationalizing assets such as pipelines and deployments, while standard pipeline implementations explicitly chain preprocessing and modeling into a single object. See IBM's documentation on pipelines and the scikit-learn pipeline documentation.

QUESTION NO: 7

Given two multidimensional arrays of the same data type, A and B which two Python NumPy statements give the matrix product of the two matrices? (Choose two.)

A. A @ B

B. A × B

C. A * B

D. np.matprod(A,B)

E. np.dot(A,B)

ANSWER: A E

Explanation:

`A @ B` performs matrix multiplication using NumPy's matrix-multiplication operator. It is equivalent to calling `np.matmul(A, B)`, applying the standard matrix-product rule to the final two axes of each array. This is the preferred concise syntax for multiplying matrices and, for arrays with more than two dimensions, it supports NumPy's batched matrix-multiplication behavior.

`np.dot(A,B)` also produces the matrix product when `A` and `B` are matrices (two-dimensional arrays): it sums products across the inner dimension, yielding the conventional linear-algebra matrix product. Thus, for compatible two-dimensional inputs, it gives the same mathematical result as the matrix-multiplication operator. NumPy documents that `dot` computes matrix multiplication for two-dimensional inputs, while the `@` operator is specifically defined as matrix multiplication through `matmul`.

Matrix dimensions must be compatible: the number of columns in `A` must equal the number of rows in `B`. See the NumPy documentation for numpy.matmul and the @ operator and numpy.dot.

QUESTION NO: 8

A data scientist is exploring transaction data from a chain of stores with several locations. The data includes store number, date of sale, and purchase amount.

If the data scientist wants to compare total monthly sales between stores, which two options would be good ways to aggregate the data? (Choose two.)

A. Find the sum of the transaction prices

B. Select the largest transaction amount by month and store

C. Write a GROUP BY query

D. Plot a time series plot of transaction amounts

E. Generate a pivot table

ANSWER: C E

Explanation:

"Write a GROUP BY query" is appropriate because the required comparison is at the combined grain of store and calendar month, rather than at the individual transaction level. A query can derive a month from each sale date, group records by store number and that month, and calculate the sum of purchase amount for every resulting group. This produces one total-sales value per store per month, which can then be compared, sorted, filtered, or visualized. IBM Db2 documents that the `GROUP BY` clause forms groups of rows for aggregate calculations such as `SUM`: IBM Db2 GROUP BY clause documentation.

"Generate a pivot table" is also a good aggregation approach. A pivot table can place stores in rows, months in columns, and purchase amount in the values area summarized as a sum. Its resulting cross-tabulated layout makes monthly totals across

locations directly comparable and supports interactive exploration, such as filtering a particular period or identifying stores with unusually high or low monthly revenue. IBM describes pivot tables as a way to organize and summarize output in tabular form: IBM SPSS Statistics pivot tables documentation. Both methods therefore create the needed store-by-month sales aggregates.