Key Takeaways

Snowflake for Data Science: Intro to Snowpark ML for Python

Task 1

- Use Snowflake UI to load CSV files,
- In case you are dealing with larger datasets, consider using other Data Ingestion methods as described in the documentation https://docs.snowflake.com/en/guides-overview-loading-data

Task 2

- Accepting Anaconda Terms and Conditions is necessary to run Python packages in your Snowflake Account.
- Hex is a powerful notebooking environment with a ready-built Snowpark Python kernel.
- You can also leverage other Python IDE : VS Code, Jupyter notebooks ..
- To create your virtual Python Environment using Anaconda follow this guide : https://docs.snowflake.com/en/developer-guide/snowpark/python/setup

Task 3

- Features preprocessing can significantly impact the performance of your model by encoding categorical variables into numerical representations that algorithms can work with effectively.
- The OrdinalEncoder function from the Snowflake.ml.modeling.preprocessing API is specifically designed for encoding categorical variables with ordinal relationships, preserving the order of categories.
- Visit Snowflake Documentation to learn more
- https://docs.snowflake.com/en/developer-guide/snowpark-ml/reference/latest/modeling#snowflake-ml-modeling-preprocessing

Task 4

- Scikit-learn pipelines do not work with Snowpark ML classes you need to use the Snowflake Version of it : snowflake.ml.modeling.pipeline package.

- By saving the preprocessing pipeline in a Snowflake stage you ensure that the same set of feature transformations and preprocessing steps are applied consistently during both the training and inference phases of your machine learning model.
- A well-structured preprocessing pipeline enhances the reproducibility of your machine learning workflow.

Task 5

- The Snowpark ML Model API currently supports sklearn, xgboost, and lightgbm models, for more details check out Snowflake Documentation : https://docs.snowflake.com/en/developer-guide/snowpark-ml/snowpark-ml-modeling
- Call .fit and .predict to train and Predict the outputs.
- Visit Snowflake Query History to see the associated queries which run inside Snowflake.

Task 6

- GridSearchCV systematically explores hyperparameter combinations to optimize your ML model.
- Specify a range of values, to conduct cross-validated searches for the best parameters.
- The best estimator reflects the highest-performing model configuration.
- A high R-squared score signifies accurate predictions, while a lower score suggests the need for investigation and model refinement.
- For more details, refer to the documentation: https://docs.snowflake.com/en/developer-guide/snowpark-ml/reference/latest/api/modeling/snowflake.ml.modeling.model_selection.GridSearchCV.

Task 7

- Saving the model as a scikit-learn (joblib) object is facilitating the model deployment and reuse in various applications.
- Vectorized User-Defined Function (UDF) is essential for efficiently applying the ML model to large datasets within Snowflake.
- Storing the UDF's results in a new Snowflake table ensures that predictions and associated data are available for further analysis, reporting, or integration with other Snowflake processes.