**1. How can learning from Study 1 enable us to identify and then exploit an opportunity in the Machine learning field? (500 words)**

The machine learning field has grown rapidly in recent years and more advancements have happened recently. Machine learning applications provide promising and reasonable solutions for many problems. There are so many opportunities in the machine learning field because of its potential and obvious outputs. This study helps us examine some of its major flaws and related issues in the software development phase. It is beneficial to know the common issues in the development because software development for machine learning requires both coding and data understanding skills. On the other hand, machine learning libraries are upgrading to solve many issues by resorting to robust functions but some practitioners don’t know about them. The developers or practitioners who are capable to handle the code smells can implement the models more precisely. In general, companies look for models which are efficient and faster than the others. Most people analyse the efficiency but do not look at the coding level problems. There is a high demand for models which are capable to address the problems with low code smells.

We can increase the efficiency of a model by taking precautions in the development phase, reduction of these issues will enable the model to stand in the industry. The code smells are the least concern while building a model, but the reduction of these smells will help the developers and companies to build more robust and structured models. Machine learning models work better while coding but due to the code smells and anti-patterns the model may become inefficient and less productive. It is very hard to find the code smells at the end of the model. This can be minimized if the developer knows about different kinds of code smells and solutions to reduce them. Code smells affects the model’s efficiency, maintainability, and complexity. The primary code smells are ‘misusing’ and ‘lack’ of knowledge about better functions. For example, the *pipeline()* function is useful for reducing the data leakage and this will help the issues downstream. Machine learning libraries provide less complex and easy-to-learn functions for solving complex problems. Exploring and implementing these functions increase the efficiency and accuracy of the model.

Using in-built pandas library functions will increase the efficiency of the code, for instance, it takes less time to execute the code. Pandas built-in functions will reduce the iterations and increase the speed. Efficient and proper memory usage plays a crucial role in machine learning models. This can be implemented with the help of different built-in functions such as *detach()*. There are many cases of misusing the functions and API’s. While implementing any functions, practitioners must take care of these parameters to avoid future problems. While handling the gradients, the developer should think about the order. Ignoring scaling will give inefficient results. Proper code smell handling can reduce the issues in the deployment phase. Controlling the generic code smells and API code smells gives us many benefits such as better performance, less error-proneness, better readability, robustness, and efficiency. So, learning these techniques is an advantage to both the practitioners and the companies.

**2. Write your thoughts about ”Opportunities implementing the process discussed in Study 1.”**

Machine learning code might generate output based on the context of the problem, but we need to identify the performance, efficiency, and understandability of the model. The model with code smells may give reasonable output with less efficiency. Many machine learning libraries provide easy solutions to these problems with in-built functions. Applying these solutions to a specific context will increase the performance and robustness of the model. Code smells are a major concern to any Machine Learning community/group. It is a concern to data scientists, machine learning application developers, machine learning library developers, code analysis tool developers, and students. This is a continuous process of learning and implementing different control methods to achieve the desired or optimum output. Implementing new and alternate functions can reduce the complexity and take less execution time. This process will help in increasing the efficiency, understandability, ease of use, and implementation of the machine learning models.

These solutions help us in dealing with the NaN values and none values. These techniques discussed above aid in better implementation of scalers and gradients in the machine learning models. These methods are even useful with memory, parameters, scaling, randomness, masking, training, evaluation, reducing data leakage, and mitigating misuse of functions. The proper use of these methods will succour Machine Learning practitioners in increasing the model's performance and reducing the execution time of the code. The major advantage of these techniques is reduced future problems, manageable, easy to learn, and time reduction. They can be used in different types of domains and for different problems. In addition, these empirical solutions are going to help data scientists and developers produce and maintain high-quality machine learning codes and thus achieve efficient results.

While doing software development for machine learning projects, we must also remind ourselves about the code smells and the related future issues that may occur in the long run. Many sources are providing promising solutions for these issues. Exploring these sources will help this machine learning developer community and data scientists in implementing code smell mitigation. The major sources for suggestions are grey literature, paper mining, and library selection. Grey literature is materials different from traditional commercial or academic publishing, for instance, online blog posts. There are many good and efficient sources for suggesting alternatives to these issues available from which more reliable functions can be studied and implemented empirically. There are also blogs written by the data practitioners which will help the students and new practitioners who have less knowledge or experience in the software development field. Sources like stack overflow and GitHub are provided with a lot of useful and practical examples and they are easily accessible publicly. From paper mining and the citations from these papers, we can gather many machine learning-specific code smells and solutions to these issues which are affecting the performance of our machine learning application models. The papers will give a deep understanding of the pitfalls a data scientist faces during the development of an application model. Finally, looking into the machine learning libraries, there are many libraries which can be used for different problems in the development of machine learning applications. Some libraries help more with specific types of issues, for example, Pandas library helps in dealing with iteration problems. Implementing these techniques on or before building a model will benefit the performance of the model.

In addition, this process is more concentrated on the code-level problems and this will also help the model in the long run. This process will also help in building robust and efficient models with machine learning coding standards. This is a continuous process to sharpen the efficiency of the code and ease the problems in the production phase.