Proceedings of the Annual General Donald R. Keith Memorial Conference
West Point, New York, USA
April 23, 2026
A Regional Conference of the Society for Industrial and Systems Engineering

Optimizing Data-Limited Image Classification Models

Carson Kellogg[1], Andrew Brigman[1], Thomas Hoyt[1], Michael Novitzky[2], and Audrey Aldridge[2]

[1]Department of Systems Engineering, United States Military Academy, West Point, New York 10996

[2]Department of Electrical Engineering and Computer Science, United States Military Academy, West Point, New York, 10996

Corresponding author's Email: carson.r.kellogg.mil@army.mil

Author Note: Carson Kellogg is a cadet at the United States Military Academy. He is a Systems Engineering Major with a focus in Artificial Intelligence and Machine Learning. He is advised by two instructors from the Department of Systems Engineering; Lieutenant Colonel Andrew Brigman, and Captain Thomas Hoyt. In addition he is advised by Dr. Michael Novitzky, an Assistant Professor and researcher from the Robotics Research Center in the Department of Electrical Engineering and Computer Science at the United States Military Academy. His assistance on this project is greatly appreciated and was instrumental in the performance of this project. The views expressed herein are those of the authors and do not reflect the position of the United States Military Academy, the Department of the Army, or the Department of War.

Abstract: This paper investigates the efficacy of adapting test plans normally leveraged for complex systems for use on a smaller scale in order to improve the accuracy of image classification models under adverse conditions, specifically a limited dataset. In order to analyze model performance and make informed changes, a test and verification plan called the LEGO Test Plan (LTP) was created based on industry examples. The LTP outlined steps for validating and verifying system performance as a classification tool. The LTP was used to assess model performance and make informed changes to training parameters in order to create a model that exceeded the LTP's performance objectives. These results not only demonstrate the efficacy of test plans on smaller-scale systems but also demonstrate how researchers can use test plans to improve data-limited image classification models.

Keywords: Test Plan, Image Classification, Machine Learning

1. Introduction

Many robotic perception systems must operate with limited labeled data, whether because target objects vary, environments change, or collecting examples is costly. Small datasets frequently produce brittle classifiers, and although large, relevant datasets can yield near-perfect models, the common remedy of "more data and more training" is often infeasible (Dishar & Muhammed, 2023). When data or compute are constrained, researchers must instead rely on tuning model parameters to improve performance. This work therefore sought to design and demonstrate a small-scale test plan, inspired by industry practices, to verify, validate, and empirically evaluate methods for enhancing an image classifier without requiring additional data or processing power.

This approach is grounded in systems engineering principles, particularly verification and validation: verification checks whether a system is built to specification, while validation ensures a system is achieves its intended purpose. The activities are typically carried out through structured test plans, but such plans are usually developed for large, complex systems. As such, selected elements from these comprehensive plans were scaled down and adapted into the LEGO Test Plan (LTP). The LTP was used to evaluate the sorting capability of four variations of the Visual Geometry Group's well-known VGG-16 image classification model (Simonyan & Zisserman, 2015) against 30 fellow classmates (referred to as *volunteers* for the remainder of this paper). Both the volunteers and the model variants sorted the same 154-piece LEGO set and were assessed on accuracy. Because the performance benchmark was to meet or exceed volunteer accuracy, the LTP guided systematic adjustments to the training pipeline and model variant parameters.

2. Background and Related Work

This project covers nearly three years of research and development on the novel Lego Sorting System (LSS), an automated sorter that ingests unsorted LEGO pieces and categorizes them into user-defined groups to accelerate inventorying. In

this work, the LSS served as a platform for collecting real-world performance data across multiple iterations of a VGG-16-based image classifier. The results were evaluated using the LTP to guide systematic adjustments and improve classification accuracy.

Test plans are structured procedures for determining whether a system satisfies user requirements under expected conditions (IEEE Computer Society, 2008). Given the novelty and limited complexity of the LSS, a custom test plan was required. The LTP drew on established verification and validation guidance from the Institute of Electrical and Electronics Engineers (IEEE) and the National Aeronautics and Space Administration (NASA) (IEEE Computer Society, 2008; Mason et al., 2012), incorporating consistent system setup, verification steps, validation activities, and block-diagram-based system descriptions to ensure comparable conditions and confirm intended functionality. The rationale for these elements is detailed in the Methodology section.

Image classification is well studied, but most models are trained under ideal, large-data conditions that rarely hold in applied settings, where data scarcity, time constraints, or limited collection opportunities impede model development. Overfitting is a common consequence of data limitation, in which models perform well on training data but generalize poorly (Sabiri, El Asri, & Rhanoui, 2022). Common mitigations include data augmentation, early stopping, and data expansion (Ying, 2019; McGuinness & O'Gara, 2019), all of which were used at stages of LSS development.

Existing work typically evaluates such techniques individually (Inoue, 2018; Wang & Perez, n.d.; Sabiri et al., 2022; Brigato, Barz, Iocchi, & Denzler, 2022). Automated optimization pipelines (e.g.,automatic machine learning) offer broader improvement (Feurer & Hutter, 2019), but generally require substantial compute and lack structured verification checkpoints. Similarly, robustness and distribution-shift benchmarks diagnose failure modes without prescribing iterative improvement (Hendrycks & Dietterich, 2019; Recht, Roelofs, Schmidt, & Shankar, 2019). Although systems engineering relies on formal verification and validation frameworks (U.S. Department of Defense, 2019; NASA, 2021), such structured methodologies are seldom applied to small-scale machine learning projects (Luckcuck, Farrell, Dennis, Dixon, & Fisher, 2019). Self-supervised pretraining can mitigate data scarcity (Chen, Kornblith, Norouzi, & Hinton, 2020) but emphasizes algorithmic strategy rather than a test-oriented process. This gap motivates the development of the LTP, a prescriptive, test-oriented framework for improving data-limited image classification models.

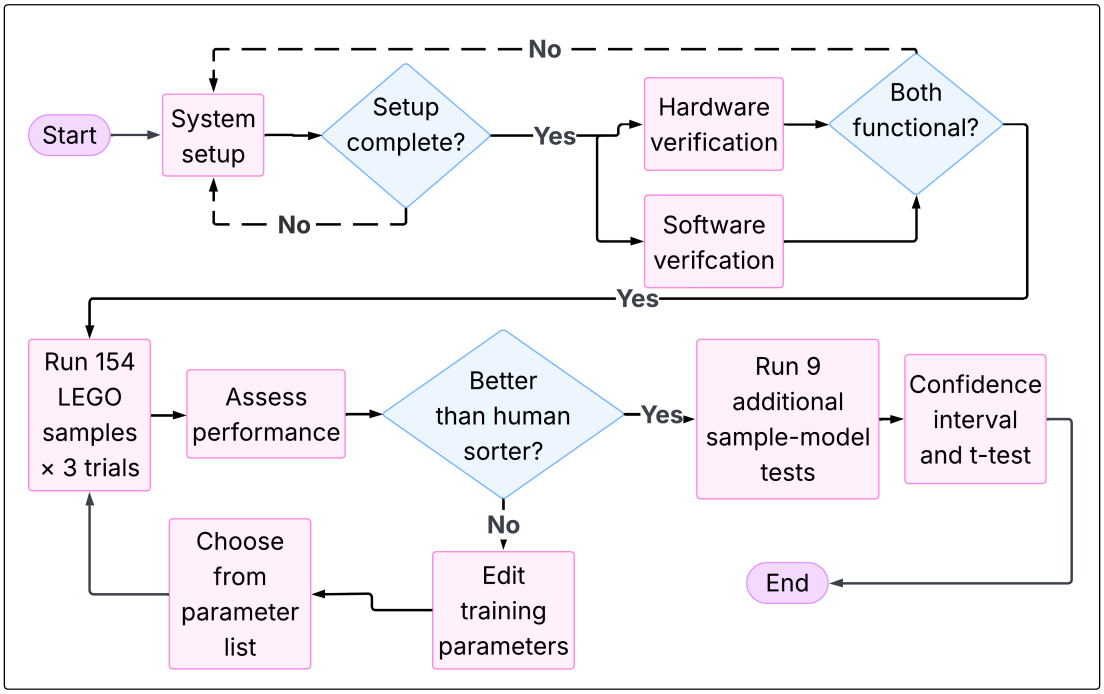

Figure 1: LTP diagram illustrating the fundamental sequence of steps and actions for setup, verification, and testing.

3. Methodology

This section summarizes the methodological framework used during testing by outlining the LTP and the rationale for each of its element. The LTP functions as a set of procedures and checks designed to ensure consistent model testing and evaluation. Its structure and decision points are shown in Fig. 1. Machine setup and verification are described only at a high level, as detailed steps are documented within the LTP itself, while model validation received expanded discussion due to its central role in the project. In Fig. 1, pink rectangles represent process nodes (groups of actions performed to achieve specific objectives), and blue diamonds represent decision nodes, where the user determines whether to proceed, restart, or modify the testing workflow.

3.1. Machine Setup

Machine setup was a comparatively minor component of the LTP and simply specified the steps required to configure the LSS. Although it involved several technical procedures, the key element was the high-level block diagram shown in Fig. 2, which clarifies the system used for data collection and to supports reproducibility of results. The block diagram depicts the LSS's three subsystems: input control, imaging, and sorting, and it illustrates how power, data, and LEGO pieces flow through components such as the hopper, conveyors, camera, and output buckets.

3.2. Machine Verification

Machine verification was adapted from NASA's test plan guidance, which emphasizes beginning with simple checks and adding complexity as needed (Mason et al., 2012). As in the machine setup section, verification was brief but essential, providing four indicators that the LSS was functioning correctly: an active video feed, movement of the conveyors and vibratory funnels, LEGO flow through the system, and the system pausing when a piece reached the imaging conveyor. Verification ensured that each data-collection iteration of began under consistent conditions and served as a natural continuation of the setup process.

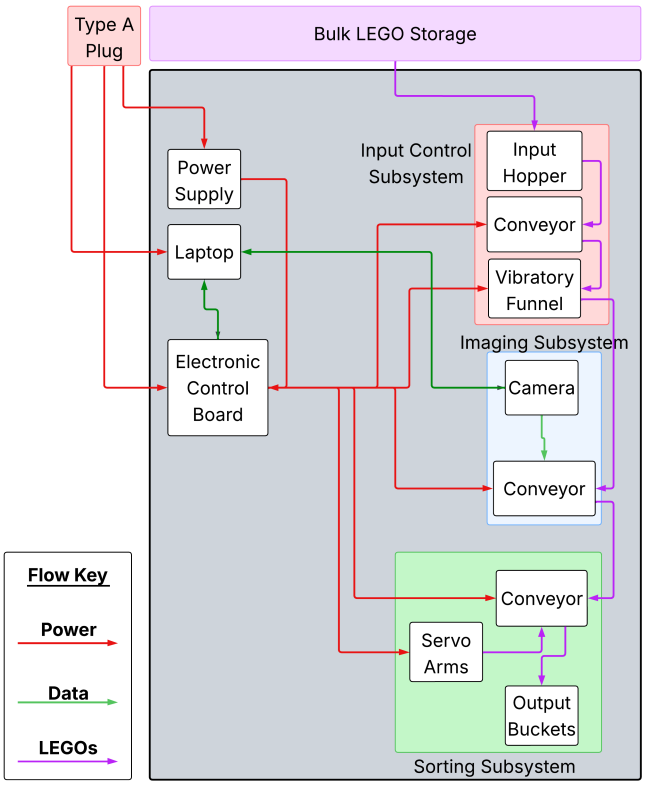

Figure 2: LSS block diagram showcasing the sub-systems and flow of power, data, and LEGOs through the system.

3.3. Model Variant Validation

Model validation guided data collection for the VGG-16 model variants and the volunteers. Each session used the same 154-piece LEGO sample, 22 pieces from each of 7 categories (plate, Studs Not on Top (SNOT), technic, brick, tile, slope, and other (Alphin, 2018)), which was mixed between trials to ensure randomness. Pieces were then fed individually into the LSS, and as each passed through the imaging subsystem, the program returned a tuple containing the part ID (e.g., 3001) and part name (e.g., Brick 1x1). A classification was considered correct if the LSS placed the piece in the proper category. Each variant was tested three times, except for the best-performing model variant, which was tested 12 times to provide a sufficient sample size for a paired t-test while limiting total data-collection effort.

4. Data Collection

Data collection occurred in two phases: human sorting and model variant testing. Human data were collected first to establish a performance baseline for the VGG-16 model variants. The human dataset included sorting times, whereas the model variant dataset did not; early trials showed that the LSS could not match human speed due to mechanical limitations. Therefore, the sorting time was removed as an evaluation criterion in the LTP.

ISBN: 97819384962-5-7

4.1. Human-Based Sorting

Thirty classmates volunteered to sort the 154-piece LEGO sample into the seven categories previously outlined. The volunteers were given a reference image of each category to ensure the data collected did not reflect a volunteer's prior knowledge of LEGO categories but rather a volunteer's sorting performance. To assess performance, task duration and accuracy were recorded. Data collection began when a volunteer entered the room and stood beside a worktable that had been prepared in advance with the sample pieces, sorting bins, and the reference sheet. The task was briefly explained to them, and any questions about the procedure were answered. However, to maintain consistency, questions about the categories or which pieces belonged to each category were not answered. Volunteers would then sort the pieces into the provided bins. Once a volunteer completed the task, their time and accuracy were recorded. Accuracy was measured by checking that each piece had been placed into the correct bin. Their time and accuracy were then entered into an excel spreadsheet for future analysis.

4.2. Machine-Based Sorting

Model data were collected on four VGG-16 variants using the same LEGO sample as the volunteers. The procedure mirrored the human trials, except that time-based measures were omitted. LEGO pieces were fed one-by-one onto the imaging conveyor, and a classification was counted as correct when the model's output matched the piece's category. Each model variant was tested three times, except the best-performing variant, which was tested 12 times to provide sufficient data for paired t-test comparisons while minimizing the overall number of required trials. The decision to test the best-performing model variant 12 times reduced the number of tests needed (from 48 to 21), resulting in a more efficient and streamlined test plan.

5. Results

5.1. Human-Based Sorting Results

Fig. 3 shows the data for all 30 volunteer sorters. There is substantial variation in both their speed anCutout, d accuracy. Accuracy ranges from 45 to 90%, while times vary from 342 s (5 min, 42 s) to 478 s (7 min, 58 s). A slight negative trend is seen, with accuracy decreasing as time increases. The most accurate volunteers also tend to complete the task more quickly. On average, volunteers were 72.4% accurate, with a standard deviation was 11.7%.

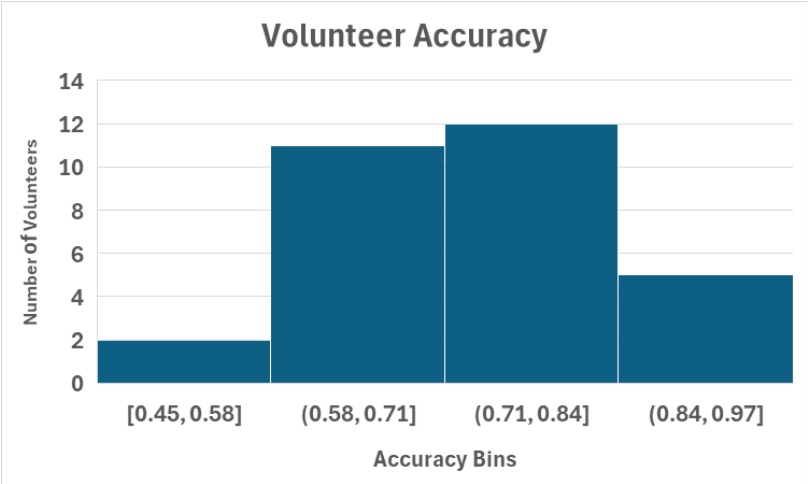

Figure 3: Human sorter performance data illustrating the variation in accuracy among volunteer sorters.

5.2. Machine-Based Sorting Results

Fig, 4 shows the comparative performance between all model variants (VGG-16, VGG-16 DA (data augmentation), New DS (VGG-16 new dataset), and NDWD (VGG-16 new dataset with weight decay), the human sorters, and a hypothetical ideal with 100% accuracy. Accuracy is characterized by LEGO category to show accuracy distribution across the different model variants. The remainder of the results section is broken down by model variant. The first model tested was a basic VGG-16 variant, changed only to accommodate the number of classes for the dataset. This model variant achieved an accuracy of 20.6%. The majority of its performance stemmed from two of the categories, specifically "plates" and "other".

5.2.1. VGG-16 Data Augmentation Results

The next model variant introduced data augmentation, as described in the background section, and achieved a modest overall accuracy increase to 26.1% over the baseline VGG-16. Accuracy improved slightly in the "plate" and "other" classes and more substantially in the "slope" class, rising from 2.4% to 5.8%. Overall improvement between the two variants was 5.5%.

5.2.2. VGG-16 New Dataset Results

Since the previous model variant had performed poorly, a sizable change to the next variant's training was required. The third variation of the VGG-16 model employed a new dataset by using images from the LSS rather than the original dataset of images scraped from Google. This resulted in the third model variant having a massive performance jump. This improvement in performance is illustrated in Fig. 4. Each piece category was classified correctly more often than not. The model variant's overall accuracy increased to 76.8%, placing it slightly above the volunteers' accuracy. Notable categories include "SNOT", "tile", and "brick", which rose from accuracy levels near zero to above 70%.

5.2.3. VGG-16 New Dataset with Weight Decay Results

The fourth and final model variant used the same data as the new-dataset variant, but additionally incorporated weight decay and early stopping in the training pipeline. The fourth variant resulted in small but meaningful gains of 3-4% accuracy. Given the success of the first three iterations of the fourth model variant, it was tested nine more times to obtain a total sample size of 12. This variant was 81% accurate. Nearly every category except for "slope" increased in accuracy during this iteration. Next, a paired t-test was performed to compare the accuracy of the volunteers with that of the fourth iteration of the VGG-16 model variant, in order to confirm statistically significant differences between the two. The t-test indicated that the accuracy of volunteers (M = 72.5%, SD = 11.7%) underperformed the fourth iteration of the VGG-16 model variant (M = 81.4%, SD = 0.0125%), $t(31) = 4.1$, $p < .05$.

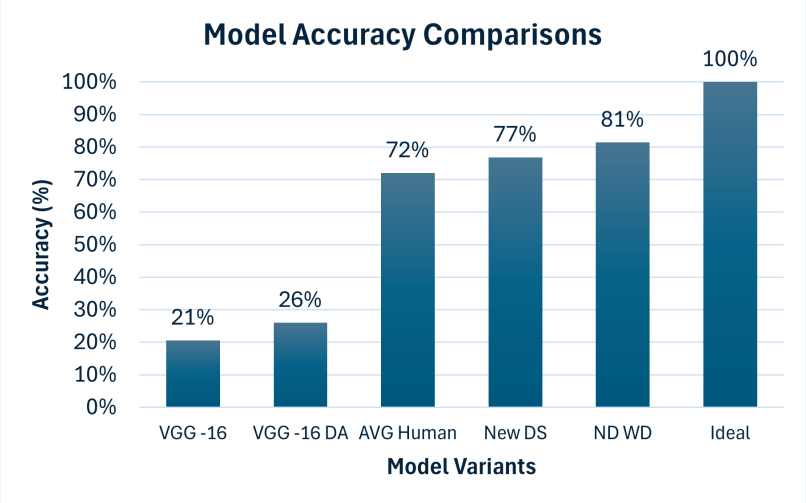

Figure 4: A stacked bar chart that compares the performance of all evaluated model variants and human sorters with each other, as well as with a hypothetical ideal benchmark of 100% accuracy.

6. Discussion

The results indicate that the LTP was effective. Each VGG-16 iteration improved in accuracy, with the largest gain (nearly 51%) occurring between the second and third iterations due to adopting a more relevant dataset. This shift allowed the model variant to focus on more meaningful image regions, improving classification performance. These adjustments followed the LTP procedures shown in Fig. 1, which also standardized LSS setup and verification. The consistent performance across categories and model variants reflects the continuity ensured by these procedures. The LTP further established clear performance benchmarks that guided parameter changes. Overall, this work demonstrates that structured test and verification plans are valuable even for simple projects, providing a framework for decision-making, reducing errors, and supporting successful project execution.

7. Conclusion

The goal of this work was to identify techniques for improving data-limited image classifiers using a test-plan methodology adapted from those used on more complex systems. Iterative testing aligned with LTP procedures and standards increased classifier accuracy from 20% to 82%, with statistical analysis indicating a 95% confidence that it outperforms human sorters in accuracy. These results demonstrate the value of structured test plans for small-scale projects and highlights steps researchers can use to strengthen their own image classification models. Although this work was limited to broad LEGO categories due to time constraints, expanding to more specific categories or all 1,914 LEGO classes could provide a clearer assessment of LTP effectiveness. Future work will increase both LTP and system complexity to further explore test-planning methods and strategies to mitigate overfitting. (National Aeronautics and Space Administration, 1970)

ISBN: 97819384962-5-7

References

Alphin, T. (2018). *Organizing your lego bricks - brick architect.* Retrieved from https://brickarchitect.com/guide/bricks/organization/

Brigato, L., Barz, B., Iocchi, L., & Denzler, J. (2022). Image classification with small datasets: Overview and benchmark. *IEEE Access*, *10*, 49233-49250. doi: 10.1109/access.2022.3172939

Chen, T., Kornblith, S., Norouzi, M., & Hinton, G. (2020). A simple framework for contrastive learning of visual representations. In *International conference on machine learning (icml)*.

Dishar, H. K., & Muhammed, L. A. (2023, 09). A review of the overfitting problem in convolution neural network and remedy approaches. *Journal of Al-Qadisiyah for Computer Science and Mathematics*, *15*. doi: 10.29304/jqcm.2023.15.2.1240

Feurer, M., & Hutter, F. (2019). Hyperparameter optimization. In *Automatic machine learning: Methods, systems, challenges* (p. 3-38). Springer.

Hendrycks, D., & Dietterich, T. (2019). Benchmarking neural network robustness to common corruptions and perturbations. In *International conference on learning representations (iclr)*.

IEEE Computer Society. (2008). *Ieee std 829™-2008 ieee standard for software and system test documentation ieee computer society.* Retrieved from https://seng.cankaya.edu.tr/wp-content/uploads/sites/53/2024/09/IEEE-Test-Doc-829-2008.pdf

Inoue, H. (2018). *Data augmentation by pairing samples for images classification.* Retrieved from https://arxiv.org/abs/1801.02929

Luckcuck, M., Farrell, M., Dennis, L. A., Dixon, C., & Fisher, M. (2019). Formal specification and verification of autonomous robotic systems: A survey. *ACM Computing Surveys*, *52*(5), 1–41.

Mason, L., Palac, D., Gibson, M., Houts, M., Warren, J., Werner, J., … Harlow, S. (2012). *Design and test plans for a nonnuclear fission power system technology demonstration unit* (Vol. NASA/TM2011217100). National Aeronautics and Space Administration. Retrieved from https://ntrs.nasa.gov/citations/20120000866

McGuinness, K., & O'Gara, S. (2019). Comparing data augmentation strategies for deep image classification. In *Imvip 2019: Irish machine vision & image processing.* doi: 10.21427/148b-ar75

NASA. (2021). *Npr 7120.5f: Nasa space flight program and project management requirements.* https://nodis3.gsfc.nasa.gov/displayDir.cfm?t=NPRc=7120s=5E. (Verification and validation planning guidance; Accessed 2026-03-11)

National Aeronautics and Space Administration. (1970, 05). *Space vehicle design criteria: Qualification testing.* NASA. Retrieved from https://ntrs.nasa.gov/api/citations/19710019569/downloads/19710019569.pdf

Recht, B., Roelofs, R., Schmidt, L., & Shankar, V. (2019). Do imagenet classifiers generalize to imagenet? In *International conference on machine learning*.

Sabiri, B., El Asri, B., & Rhanoui, M. (2022). Mechanism of overfitting avoidance techniques for training deep neural networks. In *Proceedings of the 24th international conference on enterprise information systems.* SCITEPRESS - Science and Technology Publications. Retrieved from https://www.scitepress.org/Papers/2022/111149/111149.pdf doi: 10.5220/0011114900003179

Simonyan, K., & Zisserman, A. (2015). Very deep convolutional networks for large-scale image recognition. In *International conference on learning representations (iclr)*.

U.S. Department of Defense. (2019). *Mil-std-810h: Environmental engineering considerations and laboratory tests.* https://www.dau.edu/library/DoD-Publications/Pages/Detail.aspx?pubid=MIL-STD-810H. (Accessed 2026-03-11)

Wang, J., & Perez, L. (n.d.). *The effectiveness of data augmentation in image classification using deep learning.* Retrieved from http://vision.stanford.edu/teaching/cs231n/reports/2017/pdfs/300.pdf

Ying, X. (2019, 02). An overview of overfitting and its solutions. *Journal of Physics: Conference Series*, *1168*, 022022. Retrieved from https://iopscience.iop.org/article/10.1088/1742-6596/1168/2/022022/pdf doi: 10.1088/1742-6596/1168/2/022022