# Machine Learning Campus Minden: Using Cloud-Services to teach Data Science and Deep Learning, an Experience Report

Matthias Lutz, Patrick Palsbröker, Carsten Gips and Matthias König

*Bielefeld University of Applied Sciences
{first name.last name}@fh-bielefeld.de

**Abstract**

*Within the project 'Machine Learning Campus Minden', a teaching concept was developed to further integrate the topic Machine Learning (ML) into the curriculum of the master's program in computer science at the Bielefeld University of Applied Sciences. To this end, we restructured specialized courses with ML context and created a new module 'Methods of Machine Learning'. Within this new course, all Machine Learning foundations, taught separately throughout different course modules, were centralized and built upon to allow a deeper dive into specialized topics for the more hands-on, advanced modules. These specialization modules teach students to understand demands, requirements and limitations within a real industry project on a big scale. Therefore, students explore scientific issues based on relevant data by doing group projects. If possible, cooperating companies provide the datasets. To be able to process large sets of data and fully utilize the discussed methods of Machine Learning, students are provided with cloud computing resources from three of the leading companies 'Microsoft Azure Machine Learning Studio' (MLS), 'Amazon Web Services' (AWS) and 'Google Cloud Platform' (GCP). This aggregation of cloud providers enables them to learn to integrate fitting cloud services to specific data science tasks successively. This paper discusses our experiences of creating the resulting teaching and researching concept and the lessons learned in integrating the multiple cloud service models for teaching services. It addresses both the technical and organizational needs and pitfalls and presents an evaluation regarding the cloud providers deployed in terms of domain-specific and practical problems.*

## I. Introduction

Machine Learning in all its facets gains large momentum in recent research. Thus, an enlargement of teaching felt inevitably necessary in this field. Further, an increased need for fast parallel organized hardware infrastructure has supported the ongoing demand for cloud services. Nevertheless, such a hardware infrastructure is hard to maintain as it grows with increasing demand, and it is also financially a big investment with lots of pitfalls, especially for higher education institutions [14]. Fast advancements in the field of Machine Learning increase the aging effect of computational resources, which cannot catch up with the state of the art. As of 2014, just one-third of the universities in Germany offered their computing cluster to support a wider range of projects for a deep practical experi-

*This Work was financially supported by the Federal Ministry of Education (BMBF) under the hallmark 01 IS 17081

ence with cloud technologies. So, most students do not benefit from recent computational advancements and cannot tackle problems on the scale of big data [11]. Therefore, universities should identify and control cost-effective technologies and try hard to offer realistic and reasonable access to technology for students and staff. The need for hardware and software is not being eliminated, but it is shifting from being on-premises to the cloud. In times of Industry 4.0, all information technology-related businesses, especially the small and mid-sized enterprises, will require in-house expertise of cloud resources to stay competitive [5]. This emerges from the fact that data builds the core of such enterprises, enabling various new use cases for applying Machine Learning - most efficient by the cloud. Further skills-shortage affects the whole economy; in particular smaller companies are cut back on their competitiveness. Hence, it is necessary to extend curricula considering the demand for skills required by industrial and service enterprises.

The presented concept focuses on describing an accessible multi-cloud solution for enabling unconstrained data science for teaching and research. Furthermore, a multi-cloud approach with the three major providers - as of now - provides a highly scalable cloud computing platform for schools and universities, which encompasses the three emerging essential factors to enable the faculty, students and researchers: high availability, dependability and flexibility. Additionally, all three clouds show different strengths and weaknesses, which can be exploited to build a wide range of applications and permits the students to learn what part to use in which platform and which factors qualify a platform for a certain application. A high priority factor of this project is to encourage students to adopt the use of such technologies for their future career. In order to seamlessly guide the students to use cloud services and machine learning tools, a high skill level on both sides, the students and the instructors are needed. The instructors have to cover a wide range of different tools to reflect on the domain-specific knowledge necessary to make the right choices to guide the students through theory and practice alike. The main goal is to enable the students to make feasible decisions for processing big data and make viable decisions based upon analyzing the underlying structures. They should be able to critically investigate data to identify aspects that would help to solve real-world problems and to formulate the value to the organization/research from solving them with the right use of cloud computing. Grasping the fundamental theory and concepts of cloud computing, along with the need for Machine Learning skills, will best prepare students for the entry and demanding areas of research and industry alike.

## II. Teaching Concept

The teaching and learning concept, which we developed during the project, is shown in Figure 1. The concept is hierarchically structured. It has the goal to group all preexisting and new ML modules of the master's course under one unifying roof, leading to a final master's thesis. It consists of three layers:

- 1) Base course
- 2) Advanced courses
- 3) Master thesis

The base course 'Methods of Machine Learning' provides algorithmic and methodical knowledge in Machine Learning, primarily focused on the mathematical background and formulation of the algorithms. With specially devised exercises for cloud usage, the fundamental understanding is elaborated and deepened in an individual Machine Learning project at the end of the semester. For these projects, the cloud architecture is an integral part and a project proposal must be created beforehand. Synergy effects could be observed through the concept of modules building on each other. These derive from the idea of mapping knowledge to other domains utilizing cooperating companies and bundling concise practical findings.

Based on the recent research activity at the Campus Minden, the cloud functionality was

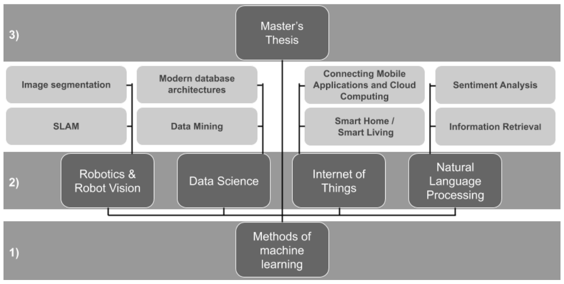


**Figure 1:** *Teaching concept*

integrated as part of the base course and advanced courses such as the ones depicted in layer two of figure 1. In the advanced courses, teaching is aligned with research and is focused on practically oriented problems, for example, brand metric comparison in the Twitter ecosystem by empirically comparing the sentimental market performance of three leading German logistics companies through an NLP pipeline or the adoption of VADER [6], a lexicon and rule-based approach for sentiment analysis with a design focus on micro-blogging content, to the German language called GerVader [13].

To use the computational power of the cloud correctly, efficient high-dimensional hyperparameter search and tuning are made possible. The newly conceptualized course modules focus on the practical usage of multiple cloud services. The base knowledge is conveyed in an initial cloud workshop. This workshop familiarizes the students with the benefits and pitfalls of each cloud service, as well as some theoretical background to each component. As there are only a few cloud-centered textbooks for teaching [11], we relied on diverse sources, for example, the online documentation and courses provided by the cloud services itself. As already described, the base course focuses on the theoretical background, followed by a small project of student's choice. Further, as the advanced courses require knowledge of cloud computing, as a further form of teaching, students work out hands-on seminars on their own. So a dependency on progress, which is reflected in the layers of the figure 1, was established in order to guarantee a high standard in the seminaristic lectures prepared by students in the advanced modules. Regarding the cloud services and providers, the students are free to choose cloud services or even combine individual components of different services to solve the task. For example, it is possible to use the Cloud Storage of GCP for persisting data like big datasets and accessing them over an HTTP request from MLS - useful to first statistical analysis - and querying the results by a GCP-service for further processing.

In this context, a crucial educational aspect is the right formulation and the access of problem-related data. Every successful ML project is based on carefully curated data, often merged accordingly from multiple sources. Thus, an aspiring data scientist should be able to form a meaningful database by asking precise questions about which fields matter and how those insights will likely matter to the cloud architecture. In terms of industry-relevant data, partnerships were formed to build projects in cooperation. Other data sources can be obtained from cooperation partners, extracted out of context fitting research projects or downloaded from online communities like Kaggle [1] offering various open datasets.

## III. Cloud Concept

As already mentioned, there were not many exhaustively evaluated works for ML oriented cloud integration with a focus on contemporary Computer Science curricula at the beginning of this project in late 2017. Thus, we identified certain fitting elements of all major cloud vendors for efficient cloud computing for Machine Learning courses on our own. The so gained knowledge was elaborated with fundamental cloud computing concepts and theory that apply to all major cloud vendors and open-source cloud platforms.

Cloud computing introduces new concepts like serverless computing as fresh ways of

[1] cf. www.kaggle.com

thinking, concerning to learn and design experiences. Serverless computing allows the execution of code, without any prior knowledge about the underlying infrastructure, which is fully managed and scaled by the cloud provider according to the payload. This degree of comfort is not always wanted and some modules require a higher level of control over the used resources. For this reason, we aimed for multiple cloud providers to allow a variety of different concepts and architectures.

In the first iteration of the new modules, we focused on the MLS and AWS as a well-balanced start in the cloud. When we have seen them successfully launched, we then completed the services with the Google Cloud Platform to cover all import aspects of the cloud providers handling important services on each service model. At this moment, we use every cloud provider on a different abstraction level: We mapped every of the three major cloud vendors to a cloud service type that we wanted to teach as follows:

- AWS: Infrastructure as a service (IaaS)
- GCP: Platform as a service (PaaS/IaaS)
- MLS: Software as a service (SaaS)

Further discussed in the next sections.

## i. Azure Machine Learning Studio

The Azure MLS represents an interactive web application. It is easily accessible for both, programmers with little experience in Machine Learning as well for experienced Machine Learning practitioners with little programming knowledge. ML pipelines are created via drag and drop to form different experiments. MLS allows for creating a flow chart with a graphical modeling interface over a selection of pre-implemented algorithms 'modules'. The modules have input and output ports to regulate the data flow between them. Depending on the module, these ports provide different output data, for example, an un- and processed data set. This structure allows for quick experiments without great inhibitions or prior knowledge.

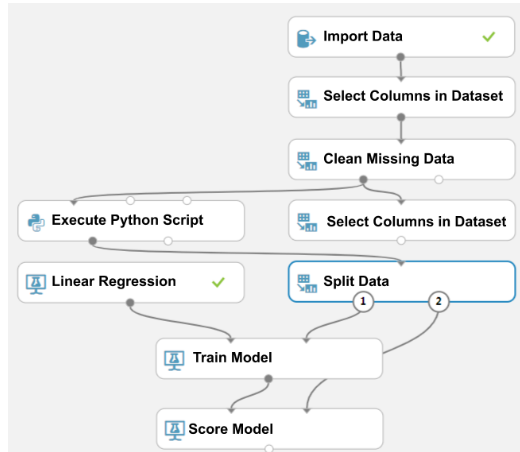


**Figure 2:** *Example of a generic ML pipeline to build a linear regression model in MLS.*

The exemplary ML pipeline in Figure 2 shows the necessary steps needed to build a linear regression model. It is evaluated afterwards based on the generalizing performance on data excluded from the training by splitting the data set. After successful training of a model, it can be directly deployed as a web service.

In cooperation with our cloud partner for MLS, a fixed monthly fee for a seat per student covered the complete functional scope. The seat was limited to seven days per executed experiment and 24 hours execution per module but also allowed for parallel processing over several nodes. So, it was naturally a viable tool to choose as a low-level entrance into the world of Machine Learning and had strengths in situations like rapid prototyping. However, it has many flaws if a flexible and configurable environment is wanted due to certain restrictions.

The platform covers many relevant application areas like recommender systems, sentiment analysis and visual search. Further, in order to provide an Internet-of-Things (IoT) compatible application for the advanced module 'Internet of Things', for example, a separate IoT hub must be created, which can be used to read out data for the MLS through consumer groups. Also, rather new advanced Machine

Learning algorithms like the long short term memory algorithm (LSTM) for classification of serial data, for instance, cannot be used. Even the scriptable component supporting Python will not allow the integration of such modern architectures. Lastly, stream processing real-time databases and applications is not supported as of November 2019.

## ii. Amazon Web Services

AWS itself provides a lot of integrated tools for a variety of different use cases, for example, IoT or mobile applications. Nevertheless, we decided not to use any of the provided management tools but to give our students access to EC2 Servers. EC2 Server on its own provides just a virtual Linux server with a GPU and pre-installed ML frameworks like Torch and TensorFlow. This way, students have the freedom as well as the responsibility to implement all their workflows and scripts.

The hourly billing model of AWS does not provide the ability to set time or cost quotas for individual machines. To keep a degree of control over the usage of the EC2 servers, we developed a web application to manage the usage of the servers and the costs that occur for every working hour. Every student is provided with a monthly quota of hours to run servers. If a students' quota is met, all his servers are shut down. Without an explicit shutdown the virtual machine (VM) would generate unnecessary costs.

## iii. Google Cloud Platform

After the successful integration of these two services, we used the newly introduced educational grants program for a complementary integration of the GCP to offer PaaS components. The grants provide educators with free usage for each student enrolled in courses with GCP as part of the curriculum. MLS was used to get a good starting point into Machine Learning, as it offered a graphical interface to easily plug components together and get a quick insight into the data. AWS was used to get a simple starting point to remote big data analytics, as it is just a configurable virtual machine (VM). With GCP, students work with more complex workflows and build Machine Learning related computation pipelines like the one shown in Figure 3. GCP offers a wide range of components provided with Cloud Identity and Access Management (IAM), so the components were dynamically unlocked as needed by the students and research staff after issuing a request, for example, for specific projects or practical tasks for obtaining the attestation for admission to the examination. This approach and defining fine granular access control Lists for cost-intensive serverless tools like Cloud Dataflow prevent misusage.

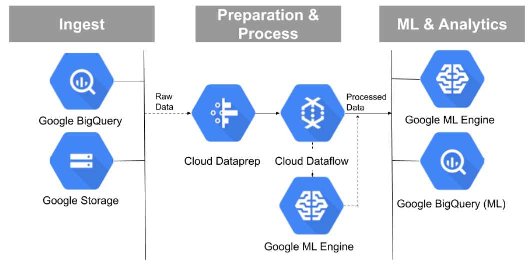


**Figure 3:** *Our designed ML pipeline for 'Methods of Machine Learning' that gets implemented throughout the course and holds as a template on relevant problems in student projects.*

The students implemented elements of this pipeline throughout attending the base module. Ranging from basic data ingestion and storage with BigQuery and Cloud Storage to more sophisticated tasks like feature engineering in Cloud Dataprep, enriching and transforming the so gained inherent knowledge on the data in a fully managed service like Cloud Dataflow. The preprocessed output by Dataflow is used to serverlessly train models in the Cloud AI services, including ML Engine, as well as to deploy it for online and batch prediction. The so implemented pipeline can flexibly switch between batch- and stream processing with Cloud Dataflow - in contrast to the MLS pipelines. Cloud Dataflow uses the well-known MapReduce algorithm at its core and is based on the Open Source Apache

Beam Distribution. It handles as one of the few available preprocessing tools batch- and stream processing without any need to reprogram.

To get familiar with all these tools, students had to solve exercises embedded as part of a short story in a Jupyter notebook - further discussed in the next chapter. Subsequently, tasks had to be mastered, which required the composition of the discussed components analogous to the ML pipeline from Figure 3. Google Cloud AI Platform APIs like AI Vision are a way to avoid retraining for basic prediction tasks and using these as a base step, for example, to manipulate a training set like extracting locations of an image dataset. Further tools for model analysis are built-in in the Data Studio and help to explain them to stakeholders and non-technical users effectively.

The Google education grants were redeemed and added at the folder level of the resource hierarchy to one single billing account, where every cost of the semester was accounted for. Stackdriver was used to set up policies to supervise the incurred expenses and predefined metrics. Exceeding the cost thresholds triggers a notification mail that is sent to the administrators. When the credits are used up, virtual machines (VMs) that are running will be stopped and no further costs will arise. Storage, including VM disks, will exist for 30 days and then be removed.

## iv. Jupyter Notebooks

As instructors, we are responsible for designing learning environments and motivational experiences, as well as teaching the hallmark of computational methodology: reproducibility [10]. To meet these expectations, we chose Jupyter notebook-based services, as these notebooks have emerged as a de facto standard for data scientists recently [2]. These notebooks are practically documents that allow executing code, display visualizations and narrative text. They can partially replace textbooks and combine them with the interactivity of an application supporting a wide range of learning goals. These goals lead to a unitary paradigm of computational thinking, comprising following learning experiences for the students [2]:

- Decomposition: Breaking down every component of the ML pipeline into smaller, manageable parts;
- Pattern Recognition: Observing patterns, anomalies and trends in data;
- Abstraction: Identifying the general principles in a particular domain that generate these patterns and formulate a generalization;
- Algorithm Design: Generalizing the approach to solve a problem into reusable instructions;

As well as a composition of cloud services complementing and enabling the full potential of these learning experiences.

Google integrated Jupyter notebooks as AI Platform Notebooks and there is also a built-in Jupyter Notebook in MLS, but the latter currently only as a preview feature. AI Platform Notebooks enables one to create and manage virtual machine instances that are prepackaged with JupyterLab. AI Platform Notebooks instances have a pre-installed suite of deep learning packages, including support for the TensorFlow and PyTorch frameworks. It is possible to configure either CPU-only or GPU-enabled instances, depending on the task. Before the use of a cloud concept with Jupyter notebooks following realization held: It was much more comfortable in many cases, to move the computer to the data than the data to the computer. Using the ingress transferring speed and the cloud-based capabilities of Jupyters' architecture, a computer is virtually addable directly to the data source. The notebook instances are protected by Google Cloud Platform (GCP) authentication and authorization as many other accessible services over HTTPS. They are accessible over a simple notebook instance URL. Notebook instances also integrate with GitHub, so that one can easily synchronize the Jupyter notebook with a GitHub and the GCP-intern cloud source repository, becoming a powerful tool.

These notebooks are quite useful in a cloud

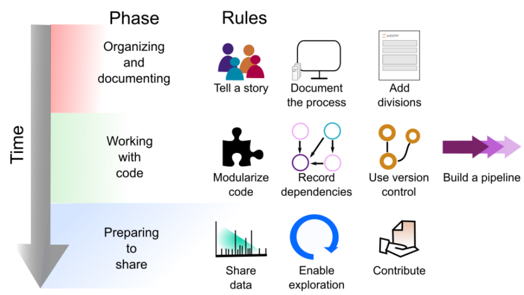


**Figure 4:** *A simplified notebook cycle incorporating the 10 rules for reproducibility suggested by [10].*

environment because of their collaborative nature and the always reproducible results by the students only needing a stable internet connection. The GCP also offers built-in connectors to other services for leveraging the full analytical toolset within the AI Platform Notebooks. In group projects, early analysis of the data and prototyping of the application can be done, accessible for anyone at any time. Lecturers can also look live into the solution process of the tasks to supervise the progress and directly help with technical problems at any time. This approach also introduces flexibility for instructor's flexibility based on the requirements and interest level of the instructors, supporting successively extendible course materials. It also eases the procedure of graded assignments, blending data, example code and mostly already installed dependencies into one place. Hence, students can focus entirely on coding and data analysis first.

### v. Reproducibility

Regarding reproducibility, the students had to comply with the 10 rules suggested by [10] for every task and project. These rules are depicted within the notebook cycle in figure 4:

While most of these phases are analogous to general software development, it is more of an art to use the underlying data to make a story of data-driven operations, as it seems that it represents the process of knowledge acquisition. One who can tell a story understands the problem at its core and fits the role of the mandatory human expert. The rest of the first phase describes the organization and documentation of processes and results. The second and third phases describe known software engineering principles like the code modularization and pipeline building. Here is only to mention that in the last step, the data should ship along with the code in a manner encouraging public explorations and contributions.

These rules must be seen as the first step to reproducibility, especially in a cloud environment, but some problems like reference breaking changes in the code library still exist. This problem is also shown in [12], who conclude that in the advent of Open Science, researchers have started to publish their research artifacts, but they lack comprehensive documentation and completeness. They stated that only three of 22 test notebooks could be successfully reproduced, meaning that code cells ran successfully and the output is equal to the initially published version. This important aspect was incorporated in assessing student learning, especially while grading and evaluating the results of projects done in the cloud.

## IV. Cloud Computing - Concerns, Risks and Solutions

The transition from on-premise infrastructure to the cloud in higher education still faces some obstacles, slowing down the adoption of cloud computing in curricula. In 2009, Gartner [3] stated that only 4% of the stakeholder from the educational sector in the USA utilized cloud computing in carrying out various educational services. In contrast to this, an interview carried out in 2010 [9] showed that 88% of the respondents agreed that there is a need to implement cloud computing in the education sector. These obstacles are primarily based on risks and financial insecurity. While financial insecurity can be handled by involving experienced consultants and using educational programs, security risks like policy and organization risks, technical risks, as well as legal risks, need long-term observation [4]. These risks emerged as unique challenges to cloud computing according to NIST [7] and remain as open issues comprising of: High degree of outsourcing, dependence on high-speed networks, sharing (multi-tenancy) and scale. The most important of these mostly known security risks include the loss of authority, lock-in, issues, isolation failure, compliance risks, management interface compromise, data protection, incomplete or insecure data deletion and malicious insiders. Hence, there is more to be done to achieve a level of maturity in terms of system security that currently exists with traditional/on-premise hosting. Higher educations should, therefore, follow best practices and countermeasures like the ones elaborated in [8], where end-to-end encryption and scanning for malicious activities can be done, for example, on the side of the university.

A factor of reliability concerning the security lies within the Service Level Agreements (SLAs) of each cloud provider, which were gradually improved along the maturity process of cloud services. These SLAs are defined in cloud agreements and establish the providers' expected uptime and performance measurements. However, they can often lack high security and privacy guarantees in the SLAs between the cloud consumers and the cloud providers, while the latter is often unilaterally favored. Also, high caution is advised when using beta features of the GCP or preview features of the MLS, for example, because they have no deprecation policy and no SLA. However, consumers are at least now mostly eligible to receive the financial credits - depending on the cloud provider, if the service level objectives defined in the specific SLAs are not met. A SLA can mostly resolve such possible disputes between consumers and providers. So typical service level agreements provided by all here discussed cloud vendors are still mostly sufficiently detailed and specific enough to meet the requirements of current higher education. Nevertheless, each cloud service has to be carefully inspected for compliance with the 'DSGVO' when using sensible data and SLAs represent a legally binding insurance policy.

Regarding our realization of each cloud provider's carefully selected services, we secured the services of each cloud provider individually. As already mentioned, for the access to the Amazon EC2 Instances the students needed to enter valid hashes in the web application, while data provided for analyses in MLS was encrypted. In the process of designing a ML pipeline on GCP, we filtered the services based on SLAs excluding beta releases and services, used and propagated best practices to the students and consulted the Google Infrastructure Security Design Overview [1] for the open issues stated above as an assurance case. The white paper describes low-level details such as physical security and secure boot, encryption of data at rest as well as communications between services and to users, keeping and keeping credentials safe. Additionally, we enforced role-based access control on every cloud platform to prevent internal employee and student threats.

Based on our experience, at least for more straightforward computations, there should exist a contingency plan. A sudden downtime or a problem at the cloud provider will not result in a loss of valuable data, standstill of work or the interruption of ongoing experiments. We suggest an iterative step-by-step integration of cloud services and vendors to mitigate potential problems further, while at the same time building at least a basic cluster to cover basic operations in case of a needed fallback. For example, shifting the workload from a standard virtual machine with its Machine Learning libraries of AWS or GCP to the self-managed cluster. The proposed cluster is only recommended for institutes with experience, as it holds new potential risks depending on its architecture.

## V. Conclusion

In this paper, we described the process and goals of integrating multiple cloud platforms into newly developed modules with a focus on Machine Learning in the master's course at the Bielefeld University of Applied Sciences. After three semesters, the new module concept has settled and even while the modules still are subject to structural optimizations, they are very well received by our students. With the integration of specially selected services of the three cloud providers on offer, we are now confident providing the right tools for each of our advanced modules. Incorporating Jupyter notebooks in our teaching by means of the AI Platform notebooks has allowed us to improve student engagement with material and their participation while presenting their solutions and to better motivate them by using more meaningful and relevant concepts. Every step of the student projects could be reproduced by executing the notebooks, which in turn processed data in the cloud.

The initial problems that we had to overcome, for example, the rental model of the cloud providers that conflicted with the buying formalities we have to comply as a public institution and to adapt all processes to the 'DS-GVO', could all been resolved. The advantages over an on-premise cluster for our use case are vast. There are no initial costs for hardware, no downtimes or maintenance, and we just pay for the time the cloud is really used. For example, we have significantly fewer costs in semester breaks. Based on our experiences, the usage of cloud platforms in teaching seems to be a viable solution. Thus, the current state of higher education should focus more intensively on emerging and industrially demanded tools to reflect current job requirements and higher output rate on practically oriented research topics, as well as guarantying reproducibility.

As future work, we need to deduce a more generalizing approach from our experiences to help faster adoption of cloud concepts with regards to data science in international curricula. A generalizing approach would also allow for a better implementation of a contingency plan. We also see a need for further investigation of ways to include tools like Qwiklabs as it offers integration for AWS and GCP and as there is also an education program offering free Qwiklabs credits for faculty members. It offers step-by-step instructions in an isolated time-limited environment, giving access to all components without the need for billing, to learn cloud concepts and also test different use cases. Furthermore, we plan to align a course about cloud computing technologies with our ML courses. This aims to provide the necessary knowledge to use any cloud provider and to understand building a cloud infrastructure, too.

## References


[1] Google infrastructure security design overview [whitepaper], 2017. Available at `https://cloud.google.com/security/infrastructure/design/resources/google_infrastructure_whitepaper_fa.pdf`.

[2] L. A. Barba, R. Talbert, A. Downey, J. Moore, D. Blank, K. E. Niemeyer, C. Willing, K. Mandli, J. Brown, L. J. Barker, and et al. Handbook for teaching and learning with jupyter, May 2019. Available at `osf.io/uqz4j`.

[3] T. Bittman. Cloud computing inquiries at gartner, 2009. Available at `https://blogs.gartner.com/thomas_bittman/2009/10/29/cloud-computing-inquiries-at-gartner/`.

[4] D. Catteddu. Cloud computing: benefits, risks and recommendations for information security. In *Iberic Web Application Security Conference*, pages 17–17. Springer, 2009.

[5] D. Foster, L. White, J. Adams, D. C. Erdil, H. Hyman, S. Kurkovsky, M. Sakr, and L. Stott. Cloud computing: developing contemporary computer science curriculum for a cloud-first future. In *Proceedings Companion of the 23rd Annual ACM Conference on Innovation and Technology in Computer Science Education - ITiCSE 2018 Companion*. ACM Press, 2018.

[6] C. J. Hutto and E. Gilbert. Vader: A parsimonious rule-based model for sentiment analysis of social media text. In *Eighth international AAAI conference on weblogs and social media*, 2014.

[7] W. Jansen and T. Grance. Sp 800-144. guidelines on security and privacy in public cloud computing. 2011.

[8] G. Ramachandra, M. Iftikhar, and F. A. Khan. A comprehensive survey on security in cloud computing. *Procedia Computer Science*, 110:465–472, 2017.

[9] N. M. Rao, C. Sasidhar, and V. S. Kumar. Cloud computing through mobile-learning. *arXiv preprint arXiv:1204.1594*, 2012.

[10] A. Rule, A. Birmingham, C. Zuniga, I. Altintas, S.-C. Huang, R. Knight, N. Moshiri, M. H. Nguyen, S. B. Rosenthal, F. Pérez, et al. Ten simple rules for reproducible research in jupyter notebooks. *arXiv preprint arXiv:1810.08055*, 2018.

[11] S. Scherzinger and A. Thor. Cloud-Technologien in der Hochschullehre – Pflicht oder Kür? *Datenbank-Spektrum*, 14(2):131–134, jun 2014.

[12] M. Schröder, F. Krüger, and S. Spors. Reproducible research is more than publishing research artefacts: A systematic analysis of jupyter notebooks from research articles. *arXiv preprint arXiv:1905.00092*, 2019.

[13] K. M. Tymann, M. Lutz, P. Palsbröker, and C. Gips. Gervader-a german adaptation of the vader sentiment analysis tool for social media texts. 2019.

[14] O. Yaghmaei and F. Binesh. Impact of applying cloud computing on universities expenses. *IOSR Journal of Business and Management*, 17(2):42 – 47, 2015.