JOINETECH

Toward Sustainable Management: Environmental and Operational Advantages of Quantum Computing over Classical HPC in the NISQ Era

Javier Sánchez-García.
Universitat Jaume I. Departamento de Administración de Empresas y Marketing.

Laura Saez-Ortuño.
Universitat de Barcelona. Business Department
Corresponding author: laurasaez@ub.edu.

Santiago Forgas-Coll.
Universitat de Barcelona. Business Department.

Ruben Huertas-Garcia.
Universitat de Barcelona. Business Department.

Citation:
Sánchez-García, J., Saez-Ortuño, L., Forgas-Coll, S., & Huertas-Garcia, R. (2026). Toward Sustainable Management: Environmental and Operational Advantages of Quantum Computing over Classical HPC in the NISQ Era. ***JOINETECH,*** 2(1), 33–46. https://doi.org/10.65479/joinetech.27

ARTICLE INFO

Keywords: Sustainability; Quantum computing; High-performance computing (HPC); Management; Marketing analytics; Quantum kernels; Energy efficiency; Environmental, social and governance (ESG); Decision-making; Noisy intermediate-scale quantum (NISQ) computing.

ABSTRACT

The urgency to decarbonize digital infrastructure motivates the search for lower-footprint computational methods in management analytics. This article evaluates, from sustainability and economic performance perspectives, the potential of near-term quantum computing [noisy intermediate-scale quantum (NISQ) computing] versus high-performance classical architectures (HPC) for management tasks such as customer classification, resource allocation, and decision optimization. We propose an evaluation framework that integrates: (i) model performance metrics (area under the curve, recall, precision), (ii) energy and carbon metrics [kWh, kilograms of CO_2 equivalent ($kgCO_2e$)] per experiment and per unit of business utility, and (iii) scalability under wall clock and queue constraints. Using a hybrid pipeline that combines quantum kernels with feature extraction and support vector machines (SVM) [quantum SVM plus quantum feature extraction (QSVM + QFE)], we observe that, for recallfirst use cases (e.g., marketing), shallowdepth circuits can maintain or improve sensitivity, enabling decisions with fewer false negatives. When classical training would be heavy (e.g., extensive hyperparameter sweeps, large kernel matrices), simulated quantum approaches or limited hardware runs can reduce total energy by requiring fewer retraining cycles and allowing receiver operating characteristic (ROC) thresholding without retraining. We present a practical measurement protocol for modern HPC infrastructures (e.g., MareNostrum 5) and outline scenarios where an environmental quantum advantage is plausible, especially with forthcoming accelerated partitions and fidelity improvements. This comparison is theoretical, based on analytical models parameterized with literature-backed ranges. We conclude with governance and environmental, social and governance (ESG) reporting recommendations and a research agenda to quantify "utility per $kgCO_2e$" for datadriven business decisions.

Submission: March 13, 2026, Acceptance: April 18, 2026. Published: May 2026.

1. Introduction

The 2030 Agenda and its 17 Sustainable Development Goals (SDGs) set out clear targets to decarbonize the digital economy, improve energy efficiency, and promote responsible innovation [United Nations, 2015; UN Department of Economic and Social Affairs (DESA), 2022]. The European Union has operationalized these objectives through the European Green Deal (European Commission, 2019), the European Climate Law (European Parliament and Council, 2021), the Fit for 55 package (European Commission, 2021), and the Artificial Intelligence Act (AI Act) (European Parliament & Council, 2024), which require transparent energy and carbon measurement as well as robust governance of advanced technologies. Spain, aligned with its Long-Term Decarbonization Strategy (Government of Spain, 2023), is promoting high-performance computing (HPC) infrastructures and efficient data centers, alongside initiatives such as the Strategic Project for Economic Recovery and Transformation (PERTE) for microelectronics and semiconductors (PERTE Chip, 2024). In this context, evaluating the value per kilogram of CO_2 equivalent ($kgCO_2e$) of data-driven decisions is not only desirable but necessary to meet key SDGs such as Affordable and Clean Energy (SDG 7); Industry, Innovation, and Infrastructure (SDG 9); and Climate Action (SDG 13) (United Nations, 2015; European Commission, 2019; European Parliament and Council, 2021).

Advanced analytics in management increasingly drives non-trivial energy consumption and associated emissions (Strubell et al., 2019; Henderson et al., 2020). State-of-the-art HPC systems such as MareNostrum 5 (MN5) deliver hundreds of peta-floating-point operations per second (PFlop/s) across general purpose and accelerated partitions [Barcelona Supercomputing Center (BSC), 2024]. However, intensive use can entail significant environmental footprints depending on occupancy, input/output (I/O), and queue dynamics [Henderson et al., 2020; EuroHPC Joint Undertaking (JU), 2025]. In parallel, quantum computing in the noisy intermediate-scale quantum (NISQ) era has advanced with methods such as quantum kernels and quantum support vector machines (Q-SVM), which realize high-dimensional embeddings via shallow circuits (Preskill, 2018; Havlíček et al., 2019; Schuld &

Killoran, 2019), potentially reducing retraining requirements while preserving decision quality (Benedetti et al., 2019; Blank et al., 2020; Cerezo et al., 2021).

This article investigates whether, and under what operational conditions, quantum computing can offer sustainability advantages for management workloads relative to classical HPC while preserving business utility. We integrate model performance with energy and carbon accounting at the experiment level, and normalized by business utility, in line with good practice and reference standards [International Organization for Standardization (ISO), 2018; Greenhouse Gas (GHG) Protocol, 2023]. Specifically, we develop a hybrid Q-SVM plus quantum feature extraction (QFE) pipeline suitable for recall-first regimes (Havlíček et al., 2019; Schuld & Killoran, 2019), propose a practical measurement protocol for HPC and NISQ devices inspired by energy-monitoring methodologies in EuroHPC and centers such as the BSC (EuroHPC JU, 2023; BSC, 2024), and delineate the conditions under which an environmental quantum advantage is plausible today versus contingent on hardware/software progress (Preskill, 2018; Cerezo et al., 2021). The measurement and reporting approach is also framed within EU regulatory instruments (the AI Act and data center sustainability) to facilitate governance and environmental, social and governance (ESG) compliance (European Parliament & Council, 2021, 2024).

In summary, this article makes a threefold, actionable contribution. First, we propose a comprehensive evaluation framework that links model performance [area under the curve (AUC), recall, precision] with energy and carbon measurement per experiment and per functional unit of business utility (utility per $kgCO_2e$/kWh), aligned with ISO 14064-1 and the GHG Protocol. Second, we present a pragmatic NISQ pipeline (Q-SVM + QFE, with Nyström and batching) that enables recall-first decisions by adjusting receiver operating characteristic (ROC) thresholds without retraining, reducing compute cycles and energy versus HPC under realistic conditions. Third, we deliver an operational measurement and reporting protocol for hybrid environments [HPC + quantum processing unit (QPU)], with observable variables (power/latencies/shots/queues) and automatic routing rules based on energy and utility breakeven points. This methodological package allows organizations to quantify and compare, in an auditable way, the value per $kgCO_2e$ of business decisions and to determine when and how quantum provides an environmental advantage today.

2. Background

Digital sustainability in analytics depends on energy consumed, grid emission factors, and hardware/software efficiency. In management contexts, environmental cost should be normalized by business utility (Sáez-Ortuño et al., 2024), for example, retained revenue associated with higher recall in retention campaigns. Modern HPC platforms combine central processing unit (CPU) and graphics processing unit (GPU) partitions, enabling broad hyperparameter exploration and large-scale simulation; however, realized efficiency depends on effective occupancy and queue behavior.

NISQ computing offers shallow circuits that evaluate overlaps in exponentially large Hilbert spaces. Recent work on quantum kernels indicates that, for certain regimes, high recall and competitive AUC are achievable with limited depth, supporting sensitivity-oriented policies and ROC thresholding without retraining (Schuld & Killoran, 2019; Huang et al., 2021). Approximation techniques such as Nyström methods can reduce complexity for kernel and feature representations while preserving downstream separability (Williams & Seeger, 2001). Energy and carbon reporting frameworks for artificial intelligence (AI) provide methodological guidance for measurement and disclosure (Henderson et al., 2020). We also reference technical documentation of MN5 for infrastructural context (Barcelona Supercomputing Center, 2024).

2.1 Regulatory and Digital Sustainability Framework

The regulatory and digital sustainability framework provides the context that justifies integrating energy and emissions measurement into management analytics. The acceleration of these practices increases energy consumption and, consequently, the carbon footprint of digital infrastructures. It is therefore essential to incorporate indicators such as kWh and $kgCO_2e$ across the lifecycle of models and decisions, aligning practice with ESG governance and tightening regulatory requirements (ISO, 2018; Greenhouse Gas Protocol, 2023; European Parliament & Council, 2024)

At the international level, the 2030 Agenda and the SDGs set clear targets for affordable and clean energy, infrastructure and innovation, and climate action (United Nations, 2015; UN DESA, 2022). Translated into analytics, this entails improving the energy efficiency of modelling processes, prioritizing infrastructures with lower carbon intensity, and reporting impacts in a verifiable manner. This ambition is underpinned by established measurement standards: ISO 14064-1:2018 sets principles for quantifying and reporting GHG emissions at the organizational level (ISO, 2018), while the Greenhouse Gas Protocol provides guidance for calculating Scope 2 and 3 emissions for IT workloads, recommending the use of location-based or market-based emission factors, phase-level breakdown (training and inference), and the definition of a functional unit to enable comparison, such as “per 1,000 predictions” (Greenhouse Gas Protocol, 2023).

The European Union has translated these objectives into operational regulatory instruments. The European Green Deal sets the ambition of climate neutrality and promotes energy efficiency and sustainable digitalization, the European Climate Law gives legal force to the 2030 and 2050 climate targets, and the Fit for 55 package directly affects the carbon intensity of electricity, shaping the factors that convert kWh into $kgCO_2e$ in HPC and QPU contexts (European Commission, 2019; European Commission, 2021; European Parliament & Council, 2021). For its part, the Artificial Intelligence Act introduces requirements for documentation, risk management, and traceability for AI systems that, although not mandating explicit energy metrics, facilitate the integration of per-experiment consumption and emissions measurements and support technical audits (European Parliament & Council, 2024). In

parallel, the EuroHPC JU ecosystem coordinates supercomputers, AI factories, and quantum computers across Europe, promoting access policies and operational best practices that foster the standardization of measurements (energy meters, CPU/GPU times, latencies) and comparability across centers (EuroHPC Joint Undertaking, 2025).

At the national level, Spain illustrates how decarbonization strategies and the 2023–2030 National Energy and Climate Plan (NECP) orient emissions reduction and the greening of the electricity mix, with a direct impact on local emission factors applied to the calculation of $kgCO_2e$ for HPC/AI workloads (Government of Spain, 2023). The push for infrastructures such as MareNostrum 5 and efficient data centers enables finer-grained measurements at node or partition level (Barcelona Supercomputing Center, 2024). In addition, the PERTE Chip strengthens the microelectronics value chain, affecting hardware efficiency and the availability of accelerators key aspects for reducing per-operation consumption and improving traceability (PERTE Chip, 2024).

For management analytics, the practical implication is twofold. First, normalization should anchor environmental impacts to business utility, enabling metrics such as "sales retained per kWh" or "false negatives avoided per $kgCO_2e$," which make classical and quantum-hybrid pipelines and different temporal cohorts comparable (Henderson et al., 2020). Second, documentation and auditing become requirements: recording per experiment hyperparameters, times, queues, latencies and number of shots, as well as orchestrator or simulator consumption; transparently converting counters and times to kWh, and kWh to $kgCO_2e$, using appropriate factors; and reporting results per homogeneous functional unit to ensure comparability (ISO, 2018; European Parliament & Council, 2024). Taken together, this regulatory and standards framework supports normalizing environmental impact by business utility and establishing comparable measurement protocols between HPC and NISQ pipelines to rigorously assess the sustainability of data-driven decisions.

2.2 Energy and Carbon Footprint in Analytics and ML

To robustly measure the environmental footprint of analytics and machine learning (ML), it is advisable to separate three aspects: the energy used in each stage (training, validation, inference, and, where applicable, data preparation and orchestration); the "cleanliness" of the electricity (by location or contractual instruments, as specified by the GHG Protocol); and hardware/software efficiency (choice of CPU/GPU, optimized libraries, parallelization, and mixed precision) (Intel, 2020; Greenhouse Gas Protocol, 2023). In classical systems, consumption is concentrated in hyperparameter sweeps, cross-validation, and the construction/handling of large matrices. In NISQ workflows, effective cost depends on job latency, the number of shots, and the overhead of the simulator or orchestrator (Schuld & Killoran, 2019; Huang et al., 2021).

In HPC, facility energy meters are combined with CPU/GPU runtimes to derive kWh; in QPU contexts, latencies, shots, and retries are recorded, reporting simulator kWh and, where hardware consumption cannot be observed, transparent operational metrics. The conversion from kWh to $kgCO_2e$ requires traceable factors and temporal and geographical context (time of day, date, electricity mix) (The Green Grid, 2012; Greenhouse Gas Protocol, 2023).

To enable comparison across projects, results should be normalized by a functional unit (e.g., per 1000 predictions) and expressed as utility per kWh or per $kgCO_2e$, as well as energy or carbon per point of recall/AUC. Adjusting the threshold via the ROC curve allows fair comparison of classical and hybrid pipelines without retraining (Henderson et al., 2020). Given uncertainties (emissions factors and limited direct metering on QPUs), ranges and proxies should be reported. Reducing problem complexity (Nyström methods, subsampling, QFE) shortens wall time and lowers energy use and emissions per decision while preserving utility in recall-first settings (Williams & Seeger, 2001).

2.3 Modern HPC Architectures and Operational Efficiency

A high-performance computer (HPC) brings together CPU- and GPU-based machines, and each is suited to different tasks. CPUs perform better when the workload has many steps and complex rules or requires extensive preprocessing, whereas GPUs excel at highly parallel, repetitive computations, such as matrix operations, achieving higher performance per unit of energy. If the task does not parallelize well, the GPU is underused and extra energy is wasted (Dongarra et al., 2020). Beyond the type of machine, interconnects between nodes and the storage system matter a great deal: When machines communicate frequently or read and write files in an irregular pattern, everything slows down and consumption rises. This is why practices such as copying data to fast local storage, batching input/output operations, and correctly binding processes to hardware help (Schmuck & Haskin, 2002; IBM, 2020; Pawlowski et al., 2021). Resource requests and queueing also matter: requesting more than you actually use leaves machines powered on but idle, and chaining multiple runs back-to-back reduces startup and coordination overheads (Yoo et al., 2003; Feitelson, 2014). Software makes a difference: Optimized libraries and lighter numerical formats, when valid, speed up computation and cut consumption, and simplifying the problem (for example, using representative samples) trims time and energy without losing useful quality (Intel, 2020; Yang et al., 2020; NVIDIA, 2023). Good measurement is key, so energy consumption and CPU/GPU times should be logged to convert them into kWh per experiment and thus derive indicators such as energy per 1000 predictions or utility per kWh (ISO, 2018; Greenhouse Gas Protocol, 2023). Finally, external factors such as data center efficiency [power unit efficiency (PUE)], how "clean" the electricity is at any given time, temperature, and queue saturation affect energy and emissions. Therefore, scheduling when electricity has a lower footprint and prioritizing experiments with higher value per kWh improve climate impact (The Green Grid, 2012; European Commission, 2021). In sum, climate effi-

ciency in HPC comes from aligning each task with the right hardware, optimizing data movement and storage, tuning software, right-sizing with care, and measuring in detail to make informed decisions (Barcelona Supercomputing Center, 2024; EuroHPC Joint Undertaking, 2025).

2.4 Foundations of NISQ computing for analytics

Current quantum computers (NISQ) feature noisy qubits and lack large-scale error correction, yet they can still deliver practical value when using simple and efficient circuits (Preskill, 2018). The central idea is to employ quantum embeddings that map classical data into spaces where they are more easily separable by simple classifiers, such as Q-SVM or QFE (Havlíček et al., 2019; Schuld & Killoran, 2019). Circuit design aims to keep depth low, typically two to five layers, using rotations to encode the data and entanglement aligned with the chip's physical topology to avoid unnecessary swaps. Measurements are performed in the Z basis with light readout mitigation, which collectively reduces the impact of noise (Harrigan et al., 2021). Quantum kernels, for their part, estimate similarity between data by computing the overlap of their quantum states (Havlíček et al., 2019; Schuld & Killoran, 2019). By contrast, QFE directly produces an explicit feature vector (for example, 64–256 dimensions) which is then processed by a classical classifier. In this way, most of the learning burden is placed on the classical side, thereby reducing the need for retraining (Schuld et al., 2020).

Unlike certain variational quantum algorithms (VQAs) that may stall during training, kernels and QFE maintain good sensitivity and competitive AUC with less computation (McClean et al., 2018; Cerezo et al., 2021). They also allow decision thresholds to be adjusted using the ROC curve without retraining, which lowers energy per decision (Fawcett, 2006).

The operational cost is concentrated in the number of repetitions ("shots"), QPU latency, and simulation time. To curb these costs, prudent techniques are used: Nyström approximation and subsampling to temper quadratic growth with dataset size, circuit batching, and adaptive shot allocation where it contributes most. These measures stabilize results and reduce consumption (Williams & Seeger, 2001; Harrigan et al., 2021). Such approaches are particularly effective with medium-sized datasets and in recall-first regimes, under queue/time constraints in HPC, with periodic validations on the QPU, where predictable fidelity and latency are advantageous (Gambetta et al., 2017).

2.5 Quantum Learning with Kernels and Quantum Support Vector Machines (Q-SVM)

Quantum learning with kernels defined as $KQ(x, x') = |\langle \psi(x)|\psi(x')\rangle|^2$ maps classical data to states $|\psi(x)\rangle$ via shallow circuits U(x) (two to five layers) to capture nonlinearities and render them separable by linear classifiers (Havlíček et al., 2019; Schuld & Killoran, 2019). Q-SVMs employ this kernel as input to a classical support vector machine (SVM) and proceed in three steps: define the encoding ansatz, estimate KQ via simulation or on the QPU with sufficient shots, and train the SVM. In this way, optimization remains on the classical side, and in recall-first scenarios, performance typically maintains a competitive AUC while improving recall (Fawcett, 2006; Schuld, 2021). The embedding design (amplitude/phase/rotation encoding and entanglement aligned with the physical topology) balances capacity and fidelity, and the "strength" of the data conditions spectral separability (Benedetti et al., 2019; Havlíček et al., 2019).

To contain costs, there are two complementary routes: (i) QFE, which measures expectation values of observables and produces explicit vectors of 64–256 features for classical classifiers, thereby avoiding the quadratic cost of the full kernel and combining well with Nyström (Williams & Seeger, 2001; Schuld et al., 2020), and (ii) a hybrid approach that simulates most of the map on HPC and reserves the QPU for validations (Harrigan et al., 2021). Once the embedding is fixed and the SVM trained, the operating point can be adjusted along the ROC via thresholds without retraining, stabilizing cycles and reducing energy (Fawcett, 2006).

NISQ noise (gate/readout errors and decoherence) is mitigated by shallow circuits, light readout calibration, and adaptive shot allocation (Gambetta et al., 2017; Preskill, 2018). Since constructing the full kernel has a cost of $O(n^2)$, one resorts to subsampling, computing only part of the matrix and completing it afterwards, or the Nyström method, which reduces the cost to $O(mn)$ (Williams & Seeger, 2001). In QFE, by contrast, the cost depends on the number of measured observables and the circuit depth (Schuld et al., 2020).

From a sustainability perspective, these choices reduce retraining and enable policy control via ROC with per-experiment energy measurement and normalization by utility (e.g., true positives per kWh, false negatives avoided per $kgCO_2e$), making an "environmental quantum advantage" plausible for medium-sized datasets, undersaturated HPC queues, non-trivial emission factors, and sufficient NISQ fidelity (Henderson et al., 2020). Therefore, quantum kernels and Q-SVMs offer a pragmatic NISQ framework with shallow embeddings, stable classical training, complexity approximations (Nyström), selective simulation, and transparent measurement to improve climate efficiency without sacrificing predictive quality.

2.6 Complexity Reduction in Kernel Methods

Kernel methods are powerful because they enable the separation of data with complex nonlinear decision boundaries (Schölkopf & Smola, 2002; Shawe-Taylor & Cristianini, 2004). The challenge lies in their computational cost since constructing the similarity (Gram) matrix for n samples requires memory that scales as n^2, and training can scale as n^3 (Williams & Seeger, 2001; Gittens & Mahoney, 2016). This affects both classical kernels [radial basis function (RBF), polynomial] and quantum kernels based on state overlaps (Havlíček et al., 2019; Schuld & Killoran, 2019). Consequently, reducing complexity is essential to maintain reasonable runtimes and energy consumption (Henderson et al., 2020).

Regarding the Nyström method, in simple terms, it avoids using all data pairs by selecting m "representative points" (m much smaller than n) and using them to approximate the full matrix (Williams & Seeger, 2001; Drineas & Mahoney, 2005). In this way, the problem's essential structure is preserved, and when the data exhibit a few dominant patterns, most of the relevant information is captured (Bach, 2013). In practice, the cost is reduced by computing $n \times m$ similarities and operating with an $m \times m$ kernel, which is much faster and lighter (Gittens & Mahoney, 2016). Choosing these points is important. Random sampling works, but selecting diverse points using techniques such as k-means (Sáez-Ortuño et al., 2023, Huertas-Garcia et al., 2025) often improves accuracy for the same m (Kumar et al., 2012). A useful procedure is to stratify the preselection, tune m with validation until the metrics cease to improve appreciably, and maintain appropriate regularization in the SVM so that the decision boundary is robust (Schölkopf & Smola, 2002).

In quantum-hybrid workflows, complexity can be reduced by estimating only those parts of the quantum kernel associated with the m representatives, thereby avoiding the computation of n^2 overlaps (Havlíček et al., 2019). With QFE, it is advisable to limit the size of the explicit feature vector (e.g., 64–256 features) and, if needed, add an approximate classical kernel via Nyström or random features so that cost grows almost linearly with n (Rahimi & Recht, 2008; Schuld et al., 2020).

There are other useful tools. Random Fourier Features project the data into D features and allow training a linear model with cost $n \times D$ (Rahimi & Recht, 2008; Le et al., 2013). Block and low-rank methods avoid forming the full matrix, relying instead on fast matrix–vector products (Halko et al., 2011). Subsampling and compression also help by selecting the most informative points or features (active sets, observable pruning in QFE, PCA/autoencoders), while maintaining separability in subsequent stages (Schuld et al., 2020).

For validation, it is advisable to plot how metrics (AUC, recall, precision) vary as m or D changes to find the "knee point" at which increasing cost yields negligible gains (Satopaa et al., 2011). Moreover, if recall is prioritized, it is worth checking that the decision threshold on the ROC curve remains stable (Fawcett, 2006).

Operationally, coordinating Nyström and related techniques with the compute scheduler and energy meters reduces CPU/GPU time, QPU calls, queueing delays, and energy consumption (Henderson et al., 2020). In large HPC systems, this translates into fewer node-hours and a smaller carbon footprint when emission factors are relevant, as well as a lower total cost of ownership (The Green Grid, 2012). Therefore, constraining m and D and systematically measuring energy per experiment and per 1000 predictions enables one to compare "full" versus "approximate" configurations, and to choose the values that maximize utility per kWh and minimize $kgCO_2e$ without sacrificing recall (Henderson et al., 2020). In this way, complexity reduction becomes a direct lever for computational sustainability.

2.7 Measurement of Energy and Carbon in HPC and QPU

Rigorous measurement of energy and carbon in hybrid environments (HPC + QPU) requires a coherent, experiment-level protocol that captures direct consumption, orchestration overheads, and applicable emission factors. The aim is to convert timings and counters into kWh, and these into $kgCO_2e$, additionally normalizing by a functional unit tied to business utility (e.g., per 1000 predictions or per decision cycle), to enable comparisons across pipelines and over time (Henderson et al., 2020). In HPC, CPU/GPU times and job states are logged; energy counters are read at the node/partition level (or estimated using average power); relevant overheads (I/O, start-ups, checkpoints) are added and converted to kWh, cross-checked against PDU/rack readings to avoid under attribution; and, where appropriate, allocation factors are incorporated (e.g., PUE or idle energy) (Henderson et al., 2020). In QPU contexts, given the usual absence of per-job kWh, traceable operational metrics are reported (job latency, number of shots, retries, and number of circuits), and the kWh of the simulator/orchestrator is measured. If the provider offers factors (energy per shot or per second), these are included, and if not, the QPU term is explicitly declared as a proxy, separating measured from estimated quantities (Henderson et al., 2020; Torlai & Melko, 2020). The conversion from kWh to $kgCO_2e$ follows location-based factors (local grid mix) and, where material, market-based factors (PPAs/guarantees of origin), documenting date/time window and energy context, and reporting ranges where uncertainties exist (counters, PUE, shot variance) (Henderson et al., 2020). Utility-based normalization presents kWh and $kgCO_2e$ per experiment and per business unit (e.g., correct positive decisions per kWh, kWh per point of recall, false negatives avoided per $kgCO_2e$), with hyperparameters frozen via a single nested cross-validation, bounded complexity (depth 2–5, QFE $\leq$ 128, Nyström $m \ll n$), and logging of queues and wall time. Transparency requires publishing hardware/software versions, scheduler configuration, measurement scripts, emission factors, and tables by functional unit to facilitate auditability and comparability (Henderson et al., 2020).

2.8 Business Utility Metrics and Normalization

The evaluation should link technical performance with business value, and both with energy and carbon. A functional unit aligned with the decision cycle is fixed (per 1000 predictions or per full data→ROC calibration→threshold cycle), and on this basis, utility indicators per kWh/$kgCO_2e$ are reported (e.g., retained sales, false negatives avoided), energy and carbon cost per performance (kWh or $kgCO_2e$ per point of recall/AUC and per 1000 predictions), and inference efficiency (ms and kWh per prediction, with its $kgCO_2e$) (Henderson et al., 2020). The protocol separates by phase (training, validation, ROC calibration, and inference) and consolidates "per cycle," uses homogeneous cohorts or weighting to correct imbalances, freezes hyperparameters once, and documents emission factors and date/time for longitudinal comparability. In recall-first contexts, the threshold is optimized on ROC/precision recall (PR) with an explicit utility function [value/costs of true positive (TP), false positive (FP), False Negatives (FN)]

to maximize utility per kWh/kgCO_2e without retraining (Fawcett, 2006). Robustness is ensured through segmentation, drift measurement and threshold sensitivity, and confidence intervals (e.g., bootstrap), always accompanied by the "complexity signature" (*n*, *m*, *D*, depth, shots, runs, and queues) to interpret efficiency. In governance, dashboards are recommended with year-on-year improvement targets in utility per kWh/kgCO_2e, limits on kWh per cycle, and a policy of not retraining unless drift thresholds are exceeded or utility degrades (Henderson et al., 2020).

2.9 Conditions for an Environmental Quantum Advantage

Environmental quantum advantage refers to situations in which a hybrid system combining quantum and classical computing matches or exceeds business value while consuming less energy per cycle and/or emitting less CO_2 per unit of utility, always under a clear and verifiable measurement method (Henderson et al., 2020). It is more likely to be achieved when the problem prioritizes capturing the largest number of relevant cases (recall-first), the decision point is adjusted via the ROC curve without the need to retrain, datasets are medium-sized, and lightweight quantum kernel or feature extraction methods are employed, supported by approximations such as Nyström or subsampling (Fawcett, 2006; Torlai & Melko, 2019). It also helps, operationally, if there are saturated HPC queues or high data ingress/egress costs, non-trivial electricity emissions factors, and QPUs with stable fidelity and latencies that allow for task batching and operation with few repetitions (shots) (Arute et al., 2019; Kim et al., 2023). A frugal design reinforces this advantage: circuits of two to five layers, QFE with up to around 128 features, Nyström with *m* much smaller than *n*, hyperparameters fixed once to avoid sweeps and retraining, and shifting adjustment to the decision threshold rather than retraining (Schuld & Killoran, 2019; Torlai & Melko, 2020).

To demonstrate this advantage, it is necessary to measure CPU and GPU consumption, as well as that of the simulation and orchestration components, to include the QPU contribution (or a transparent estimate when direct measurements are unavailable), to convert all energy to CO_2 emissions using traceable factors, and to normalize results by a unit of business utility (Henderson et al., 2020). If, when comparing by cohorts and segments, the intervals of "utility per kWh or per kgCO_2e" of the hybrid system consistently exceed those of classical alternatives, the advantage may be considered defensible (Henderson et al., 2020).

In the short to medium term, this favorable space will be expanded by improvements in QPU gate fidelity and speed, the existence of service-level agreements for queues and orchestration integrated into infrastructures such as EuroHPC, the availability of accelerated partitions with detailed energy counters that enable scheduling during lower carbon-intensity hours, and the adoption of conscious measurement practices with reuse of embeddings and thresholds to avoid unnecessary retraining (Henderson et al., 2020; Torlai & Melko, 2020; EuroHPC JU, 2023).

3. Methodology

Our objective was to compare, using equitable and transparent criteria, how much energy two technological alternatives would consume when solving typical management problems: classical high-performance computing (HPC) and NISQ-era quantum computing. Rather than running experiments on real machines—which may not be available at all times and whose behavior varies with data-center load, configuration, and time window—we chose to build analytical models. These models act as "conceptual simulations": They describe how each architecture works, which components consume energy, and how long tasks take and, from that, calculate total energy. The advantage of this approach is that anyone can review the assumptions, adjust them to their own context, and replicate the calculations (ISO, 2018; Greenhouse Gas Protocol, 2023).

The common foundation is that energy equals power times time. If a machine draws a certain power while working, total energy will be that power multiplied by the time taken. This rule applies both to a classical server cluster and to a quantum processor. The difference lies in how time is decomposed in each architecture and which parts explain power draw. In HPC, moreover, it is important to record and convert per-node power/energy readings into kWh per experiment (The Green Grid, 2012; Henderson et al., 2020).

On the quantum side, we distinguish two major time blocks. First, a "start-up" or preparation time (t_setup), which includes cooling the system to very low temperatures, calibrating quantum gates, and bringing the device to a stable state. This is a fixed cost. If we solve a single problem, that entire cost is borne by that problem; if we solve 100, the same cost is spread across the hundred. Second, the execution time (t_exec), which is spent running the circuits that encode the data and produce the required measurements. Here, circuit depth (how many layers of operations are chained), the number of measurement repetitions (shots) needed to reach a given statistical precision, and the small programming and readout latencies per shot all matter. With these elements, we approximate quantum energy as E_Q = P_Q × (t_setup + t_exec), where P_Q is the system's base power (dominated by cryogenics and control electronics). The practical consequence is that batching tasks into medium or large lots typically reduces "per-case" energy substantially because the fixed cost is diluted (Kim et al., 2023).

On the classical (HPC) side, we envisage several servers working in parallel. Each server has a characteristic power under load, and total time depends on the algorithm and how much true parallelism can be exploited. Not all time scales when adding more servers: coordination, inter-node communication and waits limit the gains. Classical energy is computed as E_C = (sum of node powers) × (algorithm time). To make the model realistic, we reflect that per-node power is not a single figure, but a range between idle and full load, and that scaling efficiency is rarely perfect. In addition, when problems require heavy information exchange or irregular memory access, communication and I/O add overheads that lengthen runtime (Yoo et al., 2003; Intel, 2020; NVIDIA, 2023).

A key aspect of the methodology was to make the fundamental technical aspects of both options explicit, without requiring advanced knowledge from the reader. On the quantum side, we outline the cooling chain, qubit quality, chip connectivity—which influences whether information must be "moved" with extra operations—and why circuit depth must be kept low in the NISQ era. On the classical side, we detail how many servers are used, their type (CPU-only or with GPUs), how they interconnect, and how the software solves the problems (e.g., number of iterations, frequency of coordination), drawing on accepted energy measurement and reporting guidance (The Green Grid, 2012; Henderson et al., 2020). With that description, we write simple formulae linking time and power to the technical elements that most drive consumption.

How does this apply to real problems? Consider planning with multiple scenarios (e.g., delivery routes under varying demand or shift scheduling with varying absences). Instead of solving a single scenario, dozens or hundreds are solved per decision cycle. This is the ideal context for quantum to amortize its start-up: If t_setup is significant but per-scenario t_exec is short and patterns repeat (same circuits, same embeddings), energy per scenario drops sharply. Conversely, if there are only a few cases, or if the formulation forces deep circuits—owing to connectivity or the problem's own structure—the advantage shrinks or disappears, and HPC is usually preferable (Dongarra et al., 2011).

To build confidence in the results, we performed sensitivity analyses. We varied parameters within reasonable ranges (e.g., more or fewer shots in quantum, more or fewer nodes in classical, latencies and powers within plausible bands) and observed how estimated consumptions change. This makes it possible to identify which factors are determinative in each architecture and to normalize results by functional unit and carbon impact in line with standards (ISO, 2018; Greenhouse Gas Protocol, 2023).

4. Results and Discussion

The theoretical comparison of energy consumption between a current-generation quantum workflow (NISQ) and classical high-performance computing (HPC) shows that the quantum option can consume less total energy under certain operational conditions, though not universally (Henderson et al., 2020; Kim et al., 2023). The difference depends mainly on how the work is structured—the batch size of instances processed per run—on the effective complexity of the quantum model, and on the level of precision required by the business and the decision timelines (Preskill, 2018; Cerezo et al., 2021).

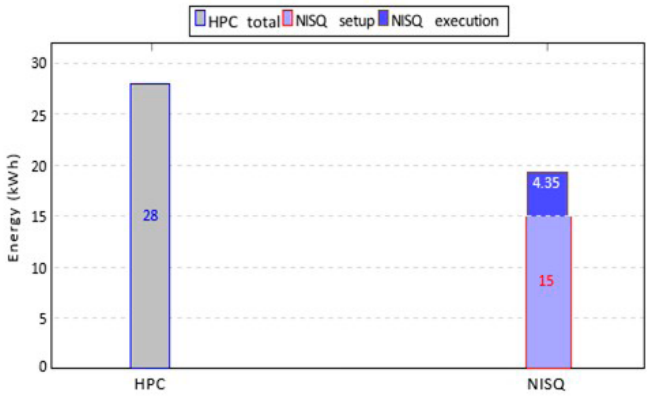
In practical terms, quantum tends to be advantageous when large batches are processed, ideally tens or hundreds of instances in a single execution, because the quantum system's preparation time, which is a fixed cost, is spread across many evaluations (Harrigan et al., 2021). In addition, the problem structure must allow quantum circuits of moderate depth; in other words, the formulation should not force many additional steps that increase the time per evaluation (Havlíček et al., 2019). Finally, when the required precision is "operational"—for example, tolerances around 1–2% on the objective rather than perfect exactness—fewer measurement repetitions are needed and execution time decreases (Fawcett, 2006; Henderson et al., 2020). Under plausible enterprise-scale parameters—quantum base power in the 20–50 kW range, setup times under an hour, per-measurement latencies of tens of microseconds, and batches of hundreds of instances—the total energy of the quantum approach can be approximately 20–40% lower than the classical alternative while meeting service deadlines (Arute et al., 2019; Barcelona Supercomputing Center, 2024). This advantage is magnified when classical solutions face scaling limits—inefficiencies due to synchronization (sync) and communication (comm) as nodes are added—or when they must deploy high-consumption clusters, for example, with multiple GPUs, to close very small optimality gaps within tight decision windows (Dongarra et al., 2011; EuroHPC Joint Undertaking, 2025).

By way of illustration, we benchmarked a planning cycle with 200 scenarios, targeting operational accuracy rather than exact optimality (approximately a 2% tolerance). We contrasted a classical approach using many servers in parallel with a nearterm hybrid quantum pipeline. In the classical setup, with 16 servers, each scenario takes about 90 s; for 200 scenarios this totals roughly 5 h and around 28 kWh of energy. In the hybrid quantum setup, the QPU requires an initial halfhour of setup, after which each scenario is processed very quickly thanks to shallow circuits and reuse of work across scenarios; the 200 scenarios then take under 40 min on the machine in total (including startup) and consume about 19 kWh. Under these assumptions, the hybrid quantum route uses roughly 31% less energy and finishes sooner. The advantage arises because batching amortizes the fixed startup and the circuits remain shallow; however, with small batches or substantially tighter tolerances and deeper circuits, the startup and perscenario time could negate the benefit. To estimate CO_2 emissions, one multiplies kWh by the appropriate location or marketbased factor (Table 1; Fig. 1).

Table 1. Energy comparison HPC versus NISQ—concrete batch (200 scenarios)

Aspect	Classical HPC	Hybrid NISQ (QFE + SVM)
Batch size	200 scenarios	200 scenarios
Power baseline	PHPC = 5.6 kW (16 × 350 W)	PQ = 30 kW
Time per scenario	90 s (including comm/sync)	≈2.62 s (128 observables × 20.48 ms)
Setup time	—	0.5 h
Total execution time	5.0 h	≈0.145 h
Total time charged	5.0 h	0.5 + 0.145 = 0.645 h
Energy model	EC = PHPC × t	EQ = PQ × (t_setup + t_exec)
Total energy	5.6 kW × 5.0 h = 28.0 kWh	30 kW×0.645 h ≈ 19.35 kWh
Relative difference	NISQ uses ~31% less energy than HPC	
Assumptions (key)	16 CPU nodes, 350 W/node, 90 s/scenario	D = 3, N_{shots} = 256.80 µs/shot, 128 observables

Fig. 1. Breakdown: NISQ energy split into setup (15.0 kWh) and execution (4.35 kWh); HPC shown as a single total (28.0 kWh)



This favorable pattern arises naturally in management contexts that, by their very nature, aggregate many scenarios per decision cycle. It occurs in fleet routing and dispatch when planning against multiple demand or traffic scenarios; in shift scheduling with structurally sparse conflict matrices and varied combinations per period; in portfolio construction with cardinality and sector limits under several risk scenarios; and in production planning with sequence-dependent setup times and uncertain demand, where numerous combinations are evaluated (Henderson et al., 2020; Cerezo et al., 2021). In all these cases, when the task-specific parameters are substituted into the breakeven condition between architectures, a quantum advantage emerges when batches reach at least tens to hundreds of instances, effective depths remain within NISQ limits, and the required precision is operational, while the classical baseline needs many nodes and exhibits sublinear efficiencies owing to communication or memory (Harrigan et al., 2021; Kim et al., 2023).

Under plausible enterprise-scale parameters—quantum base power in the 20–50 kW range, setup times under an hour, per-measurement latencies of tens of microseconds, and batches of hundreds of instances—the total energy of the quantum approach can be approximately 20–40% lower than the classical alternative while meeting service deadlines (Arute et al., 2019; Barcelona Supercomputing Center, 2024). This advantage is magnified when classical solutions face scaling limits—inefficiencies due to synchronization and communication as nodes are added—or when they must deploy high-consumption clusters, for example, with multiple GPUs, to close very small optimality gaps within tight decision windows (Dongarra et al., 2011; EuroHPC Joint Undertaking, 2025).

The quantum advantage, however, diminishes or disappears when batches are small—because the fixed setup cost is not amortized—when the problem structure forces deep circuits that lengthen evaluations, or when the required exactness is very high and drives up the number of measurement repetitions (Preskill, 2018; Cerezo et al., 2021). In these scenarios, a well-configured HPC workflow with good parallelization usually proves more energy-efficient per decision cycle (Dongarra et al., 2011; EuroHPC Joint Undertaking, 2025).

It therefore implies that, for enterprise management, quantum should be considered a batch-oriented accelerator for scenario-rich processes, not a universal replacement. In organizations with sales and operations planning (S&OP) cycles, rolling-horizon routing and scheduling, or periodic rebalancing of portfolios and capacities, structuring work to aggregate scenarios, maintaining low-complexity quantum formulations, and operating with realistic tolerances can translate into material reductions in total energy while meeting planning deadlines (Fawcett, 2006; Henderson et al., 2020). Where runs are small, the required exactness is very high, or HPC scales very efficiently, it is advisable to retain the classical route (Dongarra et al., 2011; Barcelona Supercomputing Center, 2024).

From a methodological standpoint, the results depend on explicit, auditable parameterizations: quantum power baselines, per-shot latencies, depth estimates including size weight

and power (SWAP) overhead, and statistical error targets and, on the classical side, per-node power, solver scaling curves, parallel efficiency, and communication costs (Dongarra et al., 2011; Preskill, 2018; Cerezo et al., 2021). This transparency clarifies which assumptions underpin any claimed advantage. Sensitivity analysis shows a linear dependence of quantum energy on circuit depth and shots, and a mixed linear–sublinear classical dependence on node count via efficiency; therefore, incremental hardware or algorithmic improvements can shift the breakeven point markedly (Henderson et al., 2020; Harrigan et al., 2021). In quantum, reductions in base power P_Q, faster calibrations, higher fidelities, and connectivity improvements that reduce SWAPs directly decrease energy. On the classical side, better parallel efficiency, communication-avoiding algorithms, and memory-optimized implementations reduce node-hours, often dominating (Dongarra et al., 2011).

From a risk and adoption perspective, today's quantum advantage is fragile to deviations from the favorable regime. Workload drift toward denser constraints, higher precision, or smaller batches can negate the benefit. Governance should therefore include routing rules: which instances and planning cycles go to quantum versus classical, with automated checks against the breakeven inequality (Preskill, 2018; Kim et al., 2023). Economically, energy is only one dimension; queue times, hardware availability, and operational complexity matter. If quantum access introduces scheduling delays or integration overheads, the practical benefit may diminish despite better kWh figures. Conversely, where data-center energy costs or sustainability targets are binding, even a 10–20% energy reduction on large periodic batches can be material (Henderson et al., 2020).

5. Limitations and Future Work

These findings are theoretical rather than empirical, grounded in analytical models and assumed parameter ranges rather than direct measurements on specific systems. Deviations between assumed and actual behavior—such as calibration overheads, queuing delays, operating system noise, or unexpected throttling—can materially alter outcomes. The analysis is restricted to the NISQ regime without large-scale quantum error correction; if deeper circuits or larger logical problems are required, decoherence and gate errors may invalidate the assumed time–accuracy trade-offs or necessitate substantially more shots, increasing quantum energy beyond estimates. Parameter uncertainty is significant: Quantum base power, per-shot latency, calibration cadence, gate fidelities, and classical factors such as effective parallel efficiency and communication overhead vary by vendor, firmware, workload, and time. While sensitivity analyses provide ranges, they cannot capture all operational variance. Embedding assumptions may be optimistic for some management instances; if the logical constraint graph does not embed with short paths on the device topology, SWAP overhead can increase effective depth and negate the advantage. Classical baselines may be conservative or aggressive depending on solver tuning; improved cut management, decomposition, asynchronous algorithms, and communication-avoiding implementations can reduce node-hours, narrowing the gap. Energy accounting, by design, excludes data-center overheads (cooling, facility base load) to compare only job-attributed energy; organizations with different accounting boundaries might reach different conclusions. Accuracy targets are treated as operational tolerances; if exact optimality or very tight gaps are mandated, quantum sampling demands can rise sharply, whereas if looser heuristics are acceptable on the classical side, their energy may drop. Finally, access, scheduling, and integration realities—such as limited quantum availability, application programming interface (API) overheads, or data movement costs—are not modelled and could diminish practical benefits even when theoretical energy is favorable.

To address these limitations, future work should move from purely analytical models to calibrated hybrid studies that combine controlled microbenchmarks on available quantum and classical platforms with the theoretical framework, tightening parameter ranges and validating scaling laws. On the quantum side, priority directions include modelling next-generation hardware with reduced base power and setup time, improved fidelities and mid-circuit capabilities, and richer native connectivity to curtail SWAP overhead; developing low-depth, device-aligned ansatzes and embeddings tailored to management problem structures; and advancing variance-reduction and error-mitigation schemes that lower shot counts at fixed accuracy. For classical baselines, research should incorporate state-of-the-art decomposition, asynchronous parallelism, and communication-avoiding implementations to establish stronger lower bounds on energy usage, as well as dynamic power management to reflect realistic, time-varying node power. Methodologically, extending the framework to multiobjective evaluations that balance energy, latency, and solution quality under operational constraints will better match enterprise decision contexts. Finally, end-to-end pilot deployments in scenario-rich management cycles—S&OP, rolling-horizon routing and scheduling, and portfolio rebalancing—with automated breakeven routing between quantum and classical backends will provide empirical evidence, operational insights, and data to refine the models and guide roadmap investments.

6. Conclusions

This work presents a transparent, theoretical framework to compare the energy cost of quantum and classical computing for management optimization, grounding results in explicit hardware characteristics and algorithmic assumptions. The central finding is that quantum superiority is conditional: It emerges when large scenario batches amortize cryogenic setup, effective circuit depth remains within NISQ limits so SWAP overhead is moderate, and operational accuracy tolerances keep shot counts low, particularly when classical baselines suffer poor strong scaling or require many high-power nodes to meet timelines. In plausible enterprise regimes—S&OP cycles, rolling-horizon routing and scheduling, and portfolio and production planning across many scenarios—the quantum approach can reduce total energy by roughly 20–40% while meeting decision deadlines. However, this advantage is sensitive to parameters, embedding qua-

lity, and workload structure; small batches, deep circuits, or very tight optimality targets shift the balance back to classical computing. The methodology's value lies in its reproducibility and sensitivity analysis, enabling organizations to estimate breakeven conditions and route workloads to the most energy-efficient architecture. Future progress in quantum hardware efficiency, fidelity, connectivity, and low-depth formulations, together with continued advances in classical parallel algorithms and power management, will determine how and where the frontier of energy advantage evolves in practical management applications.

References

Arute, F., Arya, K., Babbush, R., et al. (2019). Quantum supremacy using a programmable superconducting processor. Nature, 574, 505–510. https://doi.org/10.1038/s41586-019-1666-5

Bach, F. (2013). Sharp analysis of low-rank kernel matrix approximations. In S. Shalev-Shwartz & I. Steinwart (Eds.), Proceedings of the 26th Annual Conference on Learning Theory (COLT2013) (pp. 185-209).

Barcelona Supercomputing Center. (2024). Marenostrum 5: Technical information [Accessed 2025- 10-27]. https://www.bsc.es/marenostrum/marenostrum-5

Benedetti, M., Garcia-Pintos, D., Perdomo, O., Leyton-Ortega, V., Nam, Y., & Perdomo-Ortiz, A. (2019). Parameterized quantum circuits as machine learning models. Quantum Science and Technology, 4(4), 043001. https://doi.org/10.1088/2058-9565/ab4eb5

Blank, C., Park, D. K., Rhee, J.-K. K., & Petruccione, F. (2020). Quantum classifier with tailored quantum kernel. npj Quantum Information, 6, 41. https://doi.org/10.1038/s41534-020-0272-6

Cerezo, M., Arrasmith, A., Babbush, R., Benjamin, S. C., Endo, S., Fujii, K., McClean, J. R., Mitarai, K., Yuan, X., Cincio, L., & Coles, P. J. (2021). Variational quantum algorithms. Nature Reviews Physics, 3, 625–644. https://doi.org/10.1038/s42254-021-00348-9

Dongarra, J., Beckman, P., Moore, T., et al. (2011). The International Exascale Software Project roadmap. The International Journal of High Performance Computing Applications, 25(1), 3–60. https://doi.org/10.1177/1094342010391989

Drineas, P., & Mahoney, M. W. (2005). On the Nyström method for approximating a Gram matrix for improved kernel-based learning. Journal of Machine Learning Research, 6, 2153–2175.

EuroHPC Joint Undertaking. (2025). Homepage – EuroHPC JU. https://www.eurohpc-ju.europa.eu/index_en

European Commission. (2019). The European Green Deal (COM(2019) 640 final). https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=CELEX:52019DC0640

European Commission. (2021). Fit for 55: Delivering the EU's 2030 climate target. https://eur-lex.europa.eu/legal-content/EN/TXT/HTML/?uri=CELEX:52021DC0550

European Parliament & Council. (2021). Regulation (EU) 2021/1119 (European Climate Law). https://eur-lex.europa.eu/eli/reg/2021/1119/oj

European Parliament & Council. (2024, July 12). Regulation (EU) 2024/1689 of the European Parliament and of the Council of 13 June 2024 laying down harmonised rules on artificial intelligence (Artificial Intelligence Act). Official Journal of the European Union, L 2024/1689. http://data.europa.eu/eli/reg/2024/1689/oj

Fawcett, T. (2006). An introduction to ROC analysis. Pattern Recognition Letters, 27(8), 861–874. https://doi.org/10.1016/j.patrec.2005.10.010

Gambetta, J. M., Chow, J. M., & Steffen, M. (2017). Building logical qubits in a superconducting quantum computing system. npj Quantum Information, 3, 2. https://doi.org/10.1038/s41534-016-0004-0

Gittens, A., & Mahoney, M. W. (2016). Revisiting the Nyström method for improved large-scale machine learning. Journal of Machine Learning Research, 17(117), 1–65.

Government of Spain. (2023). National Energy and Climate Plan (NECP) 2023–2030. European Commission

Greenhouse Gas Protocol. (2023). Technical guidance for calculating Scope 2 and Scope 3 emissions for IT workloads. https://ghgprotocol.org

Halko, N., Martinsson, P.-G., & Tropp, J. A. (2011). Finding structure with randomness: Probabilistic algorithms for constructing approximate matrix decompositions. SIAM Review, 53(2), 217–288. https://doi.org/10.1137/090771806

Harrigan, M. P., Sung, K. J., Neeley, M., et al. (2021). Quantum approximate optimization of non-planar graph problems on a planar superconducting processor. Nature Physics, 17, 332–336. https://doi.org/10.1038/s41567-020-01105-y

Havlíček, V., Córcoles, A. D., Temme, K., Harrow, A. W., Kandala, A., Chow, J. M., & Gambetta, J. M. (2019). Supervised learning with quantum-enhanced feature spaces. Nature, 567, 209–212. https://doi.org/10.1038/s41586-019-0980-2

Henderson, P., Hu, J., Romoff, J., Brunskill, E., Jurafsky, D., & Pineau, J. (2020). Towards the systematic reporting of the energy and carbon footprints of machine learning. *Journal of Machine Learning Research*, *21* (248), 1–43.

Huang, H.-Y., Broughton, M., Mohseni, M., Babbush, R., Boixo, S., Neven, H., & McClean, J. R. (2021). Power of data in quantum machine learning. *Nature Communications*, *12* (2631), 1–9. https://doi.org/10.1038/s41467-021-22539-9

Huertas-Garcia, R., Sáez-Ortuño, L., Forgas Coll, S., & Sánchez García, J. (2024). Applying knowledge transfer in data augmentation to improve online advertising performance of entrepreneurs. Journal of Innovation & Knowledge, 10(6), 100828. https://doi.org/ 10.1016/j.jik.2025.100828

Intel. (2020). Intel oneAPI Math Kernel Library (oneMKL) documentation. https://www.intel.com/content/www/us/en/developer/tools/oneapi/onemkl-documentation.html

ISO (International Organization for Standardization), (2018). ISO 14064-1:2018 — Greenhouse gases — Part 1: Specification with guidance at the organization level for quantification and reporting of greenhouse gas emissions and removals. https://www.iso.org/standard/66453.html

Kim, Y., Eddins, A., Anand, S. et al. Evidence for the utility of quantum computing before fault tolerance. Nature 618, 500–505 (2023). https://doi.org/10.1038/s41586-023-06096-3

Kumar, S., Mohri, M., & Talwalkar, A. (2012). Sampling methods for the Nyström method. Journal of Machine Learning Research, 13, 981–1006.

McClean, J. R., Boixo, S., Smelyanskiy, V. N., Babbush, R., & Neven, H. (2018). Barren plateaus in quantum neural network training landscapes. Nature Communications, 9, 4812. https://doi.org/10.1038/s41467-018-07090-4

PERTE Chip. (2024). Microelectronics & Semiconductors: The State Enterprise for Microelectronics and Semiconductors (SEMyS) has been transformed into Spanish Society for Technological Transformation (SETT) by Royal Decree 676/2024 of 16 July. https://www.pertechip.com/home-english

Preskill, J. (2018). Quantum computing in the NISQ era and beyond. Quantum, 2, 79. https://doi.org/10.22331/q-2018-08-06-79

Rahimi, A., & Recht, B. (2008). Random features for large-scale kernel machines. In J. C. Platt, D. Koller, Y. Singer, & S. Roweis (Eds.), Advances in Neural Information Processing Systems 20 (NeurIPS 2007) (pp. 1177–1184). Curran Associates.

Sáez Ortuño, L., Huertas Garcia, R., Forgas Coll, S., & Puertas-Prats, E. (2023). How can entrepreneurs improve digital market segmentation? A comparative analysis of supervised and unsupervised learning algorithms. International Entrepreneurship and Management Journal, 19(4), 1893-1920. https://doi.org/10.1007/s11365-023-00882-1

SáezOrtuño, L., HuertasGarcia, R., ForgasColl, S., & SánchezGarcía, J. (2024). Quantum computing for market research. Journal of Innovation & Knowledge, 9(3), 100510. https://doi.org/10.1016/j.jik.2024.100510

SáezOrtuño, L., ForgasColl, S., & Ferrara, M. (2025). Quantum kernel methods: Convergence theory, separation bounds and applications to marketing analytics. arXiv. https://doi.org/10.48550/arXiv.2510.11744

Satopaa, V., Albrecht, J., Irwin, D., & Raghavan, B. (2011). Finding a "kneedle" in a haystack: Detecting knee points in system behavior. 2011 31st International Conference on Distributed Computing Systems Workshops (ICDCSW) (pp. 166–171). IEEE. https://doi.org/10.1109/ICDCSW.2011.20

Schölkopf, B., & Smola, A. J. (2002). Learning with kernels: Support vector machines, regularization, optimization, and beyond. MIT Press.

Schuld, M., Bocharov, A., Svore, K. M., & Wiebe, N. (2020). Circuit-centric quantum classifiers. Physical Review A, 101(3), 032308. https://doi.org/10.1103/PhysRevA.101.032308

Schuld, M., & Killoran, N. (2019). Quantum machine learning in feature hilbert spaces. *Physical Review Letters*, *122* (040504), 1–7. https://doi.org/10.1103/PhysRevLett.122.040504

Shawe-Taylor, J., & Cristianini, N. (2004). Kernel methods for pattern analysis. Cambridge University Press.

Strubell, E., Ganesh, A., & McCallum, A. (2019). Energy and policy considerations for deep learning in NLP. Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, 3645–3650. https://aclanthology.org/P19-1355

The Green Grid. (2012). PUE: A comprehensive examination of the metric. https://www.thegreengrid.org/en/resources/library-and-tools/20-PUE%3A-A-Comprehensive-Examination-of-the-Metric

Torlai, G., & Melko, R.G. (2019). Machine-Learning Quantum States in the NISQ Era. Annual Review of Condensed Matter Physics, 11, 325-344. https://doi.org/10.1146/annurev-conmatphys-031119-050651

UN DESA. (2022). The Sustainable Development Goals Report 2022. https://unstats.un.org/sdgs/report/2022/

United Nations. (2015). Transforming our world: The 2030 Agenda for Sustainable Development (A/RES/70/1). https://sdgs.un.org/2030agenda

Williams, C. K. I., & Seeger, M. (2001). Using the Nyström method to speed up kernel machines. In T. K. Leen, T. G. Dietterich, & V. Tresp (Eds.), Advances in Neural Information Processing Systems 13 (pp. 682–688). MIT Press.

Annex

Glossary of terms

Term	Acronym	Definition	Section/context
2030 Agenda	—	Global UN framework with 17 SDGs for 2015–2030	Introduction; 2.1
AI Act	AI Act	EU Regulation 2024/1689 on AI systems (documentation, risk, traceability)	Introduction; 2.1
Algorithmic time	t_alg	Classical solution time (function of size/density, η)	Methodology
Amortization of fixed costs	—	Allocation of a fixed cost (e.g., t_setup) across multiple runs/batches	Methodology; Results
Ansatz (quantum)	—	Circuit family/structure used to parametrize states/embeddings	Methodology; 2.4–2.5
Area under the ROC curve	AUC	Performance metric summarizing a classifier's ability to distinguish classes	Abstract; 2.8
Arbitrary waveform generator	AWG	Electronics for generating control pulses for quantum gates	Methodology
Base quantum power	P_Q	Steady power (He3/He4 cryogenics and control electronics)	Methodology
Batch	—	Set of instances/scenarios processed in a single run	Results
Calibration (QPU)	—	Periodic adjustments of gates and readout on the QPU	Methodology
Central processing unit	CPU	General-purpose computing hardware in HPC	2.3; Methodology
Circuit depth	D_circuit	Effective number of sequential layers/gates in a circuit	Methodology; 2.4
Classical energy	E_C	Total energy on HPC: E_C = (Σ P_node) × t_alg	Methodology
Constraint programming	CP	Paradigm for solving combinatorial problems	Methodology
Data center energy efficiency	PUE	Metric: total facility energy/IT energy	2.3; 2.7
Decision threshold	—	Operating point on ROC/PR to balance precision and recall by utility	2.5; 2.8
Dynamic random-access memory	DRAM	Main high-capacity memory in nodes	2.3; Methodology
Emission factor	—	Converter from kWh to $kgCO_2e$ according to electricity mix (location/market)	2.1–2.2; 2.7
ESG	—	Environmental, social and governance criteria for practice and reporting	Abstract; 2.1
Execution time (QPU)	t_exec	Net time to solve a batch on the QPU	Methodology
Functional unit	—	Normalization basis (e.g., per 1000 predictions)	2.1–2.2; 2.7–2.8
Gate fidelity	—	Probability that a quantum gate operates correctly	Methodology; 2.4
GHG protocol	—	Guidance for calculating/reporting emissions (Scopes 2 and 3)	2.1–2.2; 2.7
Gram matrix	—	Similarity matrix in kernel methods; cost $O(n^2)$	2.6
Graphics processing unit	GPU	Hardware for massive parallel computation (matrices)	2.3; Methodology
High-bandwidth memory	HBM	Memory used in accelerator (GPU) nodes	2.3; Methodology
High-performance computing	HPC	CPU/GPU clusters with high-speed interconnects	General; 2.3
Interconnect	—	Network between nodes (topology, bandwidth, latency)	2.3; Methodology
IQ mixer	—	Analogue radio frequency (RF) electronics for microwave signals in qubit control	Methodology
Ising/Quadratic Unconstrained Binary Optimization	Ising/QUBO	Binary optimization formulations used in quantum mappings	Methodology
Job queue	—	Scheduling and waiting in HPC/QPU	2.3; 2.7
Kilograms of CO_2 equivalent	$kgCO_2e$	Unit expressing greenhouse-gas emissions as CO_2 equivalents	Abstract; 2.1–2.8

Term	Acronym	Definition	Section/context
Kilowatt-hour	kWh	Unit of electrical energy	Abstract; 2.2; 2.7
Mixed-integer linear programming	MILP	Optimization with integer and continuous variables	Methodology
Noisy intermediate-scale quantum era	NISQ era	Intermediate-scale quantum computing with noise and no large-scale error correction	General; 2.4
Nyström method	—	Kernel-matrix approximation using m representative points ($m \ll n$)	2.6
Optimality gap	—	Relative difference between the best-known solution and the theoretical bound	Methodology; Results
Parallel efficiency	η	Scaling efficiency as nodes increase	Methodology
Penalties (QUBO/Ising)	—	Terms enforcing constraints via the objective function	Methodology
PERTE Chip	—	Spanish initiative to strengthen microelectronics and semiconductors	2.1
Post-processing and error mitigation	—	Techniques to reduce bias/variance without full error correction	Methodology; 2.4
Precision	—	True positives/predicted positives	Abstract; 2.8
Precision–recall curve	PR	Relation between precision and recall; useful for imbalanced classes	2.5; 2.8
Programming time	T_program	Latency to load/program the circuit	Methodology
Processing time	T_proc	Demodulation/processing latency after readout	Methodology
Quantum approximate optimization algorithm/variational quantum eigensolver	QAOA/VQE	Variational algorithms for optimization/eigenstates	Methodology; 2.4
Quantum embedding	—	Mapping data to a Hilbert space via circuit U(x)	2.4–2.5
Quantum energy	E_Q	Total energy on QPU: E_Q = P_Q × (t_setup + t_exec)	Methodology
Quantum feature extraction	QFE	Produces explicit vectors (e.g., 64–256) from observables	Abstract; 2.4–2.5
Quantum gate	—	Elementary operation on qubits (average duration t_gate_avg)	Methodology; 2.4
Quantum kernel	K_Q	Similarity via state overlap	□ψ(x)
Quantum processing unit	QPU	Qubit processor (e.g., superconducting transmon)	2.4; Methodology
Readout time	T_readout	Measurement window (readout) of qubit states	Methodology
Recall (sensitivity)	—	True positives captured/actual positives	Abstract; 2.8
ROC curve	ROC	Relation of the true positive rate (TPR) versus false positive rate (FPR) as the threshold varies	Abstract; 2.8
Sales and Operations Planning	S&OP	Sales and operations planning (periodic cycles)	Results; Conclusions
Scheduler	Scheduler	Orchestrator that allocates resources and manages queues	2.3; 2.7
Sensitivity analysis	—	Study of result variation under changes in dominant parameters	Methodology; Results
Setup time	t_setup	Cryogenic stabilization and system initialization/calibration	Methodology
Shots (repetitions)	—	Number of measurements to estimate expectations at target error	Methodology; 2.4
Strong/weak scaling	—	Efficiency with fixed/increasing problem size as nodes are added	2.3; Methodology
Size weight and power gate	SWAP gate	Operation to move states between neighboring qubits	Methodology; 2.4
T1/T2 coherences	T1/T2	Qubit relaxation and dephasing times	Methodology; 2.4
Time window	—	Date/time interval for emission factors and measurement	2.7
Topology (connectivity)	—	Physical layout [e.g., two-dimensional (2D) mesh] conditioning routes and SWAPs	Methodology; 2.4
True positive/ false positive/ false negative/ true negative	TP/FP/FN/TN	True/false positives and negatives	2.8

Term	Acronym	Definition	Section/context
Transmon	—	Type of superconducting qubit used in NISQ computing	Methodology; 2.4
Utility unit (utility per kWh/ $kgCO_2e$)	—	Business value per energy/emissions (e.g., FNs avoided per $kgCO_2e$)	2.8; 2.9
Variance reduction	—	Strategies to decrease shots while maintaining target precision	Methodology; 2.4
Wall-clock time	—	Real elapsed time including queues and overheads	2.3; 2.7
Z-basis readout	—	Measurement in the computational basis of qubits	2.4