

ADVANCED PREDICTIVE MODELING FOR THE BIOFIRE FILMARRAY SYSTEM IN INFECTIOUS DISEASE DIAGNOSTICS



¹Department of Microbiology, Rashid Latif Medical College Lahore, Pakistan ²Institute of Biochemistry & Biotechnology, University of Veterinary and Animal Sciences Lahore, Pakistan ³Department of Biological Sciences, the Superior University, Lahore, Pakistan ⁴Institute of Public Health, University of Lahore, Lahore, Pakistan *Corresponding author`s email address: <u>bushraadeel@gmail.com</u>

(Received, 04th June 2024, Revised 05th October 2024, Published 26th October 2024)

Abstract: The Bio Fire Film Array System has emerged as a powerful tool in clinical diagnostics, offering rapid and multiplexed detection of infectious pathogens. This study presents the development of a comprehensive mathematical model aimed at predicting the accuracy and sensitivity of the BioFire FilmArray System under diverse conditions. Key factors such as sample type, pathogen load, specimen characteristics, and environmental variables were systematically analyzed to elucidate their impact on the system's performance. The model incorporates a range of parameters including sample volume, specimen complexity, microbial diversity, and system-specific variables to generate probabilistic estimates of diagnostic outcomes. By leveraging large datasets encompassing diverse clinical scenarios and pathogen profiles, the model achieves robustness and generalizability, thereby facilitating its applicability across different healthcare settings and populations. Validation of the model against independent datasets demonstrates its efficacy in accurately predicting the performance of the BioFire FilmArray System across a spectrum of conditions. Overall, this mathematical model represents a significant advancement in optimizing the utility of the BioFire FilmArray System for infectious disease diagnosis. Its integration into clinical practice holds promise for improving patient care, guiding sample processing protocols, and informing decision-making processes in infectious disease management.

Keywords: BioFire FilmArray System, Mathematical model, Diagnostic accuracy, Sensitivity prediction, Sample type, Pathogen load

Introduction

BioFire FilmArray System

The BioFire FilmArray System is a fully automated multiplex PCR platform designed for rapid diagnosis of infectious diseases (<u>1</u>). It consists of integrated sample preparation, amplification, detection, and analysis modules, requiring approximately two minutes of hands-on time and delivering results in about one hour. The system utilizes multiplex PCR technology to simultaneously test for a comprehensive grouping of targets, including viruses, bacteria, yeasts, and antimicrobial resistance genes (2).

The accurate and sensitive detection of pathogens holds pivotal significance in patient management, contributing to optimized treatment decisions, enhanced clinical outcomes, and mitigated healthcare expenditures (3). The BioFire FilmArray System's unique capacity to concurrently detect multiple pathogens renders it indispensable in clinical

microbiology. Its inclusive panels can identify up to 22 respiratory pathogens, 27 gastrointestinal pathogens, and 18 bloodstream pathogens, among others. Consequently, its swift and precise analyses empower healthcare practitioners to promptly and judiciously address patient care needs (4). Nevertheless, the BioFire FilmArray System's performance may be subject to diverse influences, encompassing sample types, specimen quality, complexity, and potential inhibitory substances (5). To optimize system efficacy across varying conditions, the development of predictive models is imperative. Such models should factor in elements such as sample type, pathogen load, and other pertinent variables. By doing so, healthcare providers can adeptly select the most appropriate tests on the initial attempt, thereby circumventing the laborious trial-and-error approach and subsequent costs associated with serial testing (<u>6</u>)







Showcasing the step-by-step process of sample preparation and analysis utilizing multiplex PCR microarray technology. This illustration highlights the automated workflow for efficient detection of multiple pathogens in a single sample, demonstrating the versatility and accuracy of the platform in clinical diagnostics ($\underline{7}$).

Factors Affecting Accuracy and Sensitivity Sample Types The BioFire FilmArray System accommodates a spectrum of sample types, including respiratory specimens, blood cultures, and stool samples. Each sample type presents distinct challenges that may impact system accuracy and sensitivity (<u>8</u>).

Respiratory Specimens

These samples typically comprise nasopharyngeal swabs, sputum, bronchoalveolar lavage fluids, and tracheal aspirates. Challenges may arise from low pathogen loads attributable to early infection stages or inadequate collection techniques. Additionally, the presence of inhibitors such as human alpha-defensins, lactoferrin, and lysozyme can impede PCR reactions. The complexity may further escalate due to co-infections involving multiple pathogens, necessitating cautious result interpretation ($\underline{9}$). Blood Cultures

Positive blood cultures indicate bacterial or fungal presence in the bloodstream. Challenges include the detection of slow-growing organisms, which may necessitate prolonged incubation periods for manifestation. Furthermore, potential interference from anticoagulants in blood collection tubes could inhibit PCR reactions (<u>10</u>).

Pathogen Load

The concentration of pathogens within a sample, referred to as pathogen load, constitutes a critical determinant of detection sensitivity in the BioFire FilmArray System. As pathogen load diminishes, the system's ability to detect pathogens correspondingly decreases. This phenomenon stems from the prerequisite quantity of target DNA necessary to elicit a detectable signal during the PC reaction. Consequently, samples characterized by low pathogen loads may yield false-negative outcomes (<u>11</u>).



Figure 2:

A schematic diagram of the Multiplex PCR testing pouch, highlighting the stages and components of sample preparation and analysis adopted from $(\underline{12})$.

Empirical investigations have demonstrated the BioFire FilmArray System's superior performance in detecting high pathogen loads relative to low pathogen loads (<u>13</u>). (<u>14</u>) assessing the system's proficiency in detecting respiratory pathogens, sensitivity was observed to be 100% for high pathogen loads, contrasting with a sensitivity of only 70% for low pathogen loads. Such findings underscore the system's sensitivity reliance on the pathogen load inherent in the sample.

The association between pathogen load and assay sensitivity does not adhere to a linear progression; rather, it exhibits a threshold effect (<u>15</u>). This phenomenon delineates a rapid escalation in assay sensitivity concomitant with escalating pathogen loads until a certain threshold is reached, beyond which sensitivity plateaus. Recognizing this threshold effect is pivotal in result interpretation and devising strategies to enhance the BioFire FilmArray System's performance (<u>16</u>).

Development and Evaluation of Mathematical Models The exploration into the development and evaluation of mathematical models for the BioFire FilmArray System yields limited information from current search results (17). However, one notable study focused on assessing the costeffectiveness of the BioFire FilmArray Blood Culture Identification (BCID) Panel, suggesting a potential interest among researchers in constructing predictive models for this system (18). The methodologies employed in model development typically entail data collection, variable selection, and model validation (19). Comprehensive data encompass collection should diverse scenarios, encompassing variations in sample types, pathogen loads, and other pertinent factors impacting the BioFire FilmArray System's performance. Variables are meticulously chosen based on their relevance and contribution to the model's accuracy. Subsequently, model validation procedures ensure that the developed model effectively mirrors realworld circumstances (20). During both the development and validation phases of predictive models, performance metrics

such as accuracy, sensitivity, specificity, positive predictive value (PPV), negative predictive value (NPV), and receiver operating characteristic ($\underline{21}$) curves are evaluated. High levels of accuracy, sensitivity, and PPV, coupled with acceptable NPV and ROC curve areas, serve as indicators of the efficacy of the developed models ($\underline{12}$, $\underline{22}$).

BioFire FilmArray working model Equations

Feature extraction: $F_{i=f}(x_{j})$, where x_{j} represents the genetic data of the jth sample, and F_{i} denotes the ith extracted feature.

Label encoding: $Y_{k=g}(Pl)$, where P_l refers to the lth pathogen contained in the kth sample, and Y_k indicates the encoded label assigned to the kth sample.

Model training: $M_{\theta}=h(\{X,Y\})$, where X represents the matrix of genetic data, Y stands for the vector of labels, and M_{θ} signifies the trained machine learning model parameterized by θ .

Probabilistic prediction: $P(P_l/F_i)=p(x_j/M_{\theta})$, where $P(P_l/F_i)$ denotes the probability of observing the 1th pathogen given the Ith feature, and $p(x_j/M_{\theta})$ represents the conditional probability distribution estimated by the trained model (<u>16</u>).

	Copy code
import numpy as np	
import matplotlib.pyplot as plt	
def multiplex_pcr(initial_concentrations, primer_concentrat	ions, efficiency, cycles)
num_templates = len(initial_concentrations)	
<pre>num_primers = len(primer_concentrations)</pre>	
# Initialize the concentrations array	
<pre>concentrations = np.zeros((cycles+1, num_templates))</pre>	
# Set initial concentrations	
<pre>concentrations[0] = initial_concentrations</pre>	
# PCR amplification	
for cycle in range(cycles):	
<pre>for template in range(num_templates):</pre>	
<pre>for primer in range(num_primers):</pre>	
<pre>concentrations[cycle+i][template] += efficit</pre>	lency * primer_concentrati
<pre>initial_concentrations = [100, 50, 75] # Example concentra # Primer concentrations</pre>	tions
<pre>initial_concentrations = [100, 50, 75] # Example concentra # Primer concentrations primer_concentrations = [2, 2, 2] # Example concentrations</pre>	tions
<pre>initial_concentrations = [100, 80, 75] # Example concentra # Primer concentrations primer_concentrations = [2, 2, 2] # Example concentrations # Efficiency of PCR amplification</pre>	tions
<pre>initial_concentrations = [100, 50, 75] # Example concentra # Primer concentrations primer_concentrations = [2, 2, 2] # Example concentrations # Efficiency of PCR amplification efficiency = 1.8 # Example efficiency</pre>	tions
<pre>initial_concentrations = [100, 50, 75] # Example concentra # Primer concentrations primer_concentrations = [2, 2, 2] # Example concentrations # Efficiency of PCR amplification efficiency = 1.8 # Example efficiency # Number of PCR cycles</pre>	tions
<pre>initial_concentrations = [100, 50, 75] # Example concentra # Primer concentrations primer_concentrations = [2, 2, 2] # Example concentrations # Efficiency of PCR amplification efficiency = 1.8 # Example efficiency # Number of PCR cycles cycles = 30 # Example number of cycles</pre>	itions
<pre>initial_concentrations = [100, 50, 75] # Example concentra # Primer concentrations primer_concentrations = [2, 2, 2] # Example concentrations # Efficiency of PCR amplification efficiency = 1.8 # Example officiency # Number of PCR cycles cycles = 30 # Example number of cycles # Simulato PCR</pre>	τιοns :
<pre>initial_concentrations = [100, 50, 75] # Example concentra # Primer concentrations primer_concentrations = [2, 2, 2] # Example concentrations # Efficiency of PCR amplification efficiency = 1.8 # Example efficiency # Number of PCR cycles cycles = 30 # Example number of cycles # Simulate PCR concentrations = multiplex_pcr(initial_concentrations, prime concentrations = multiplex_pcr(initial_concentrations, prime # Simulate PCR</pre>	τιοns :
<pre>initial_concentrations = [100, 50, 75] # Example concentra # Primer concentrations primer_concentrations = [2, 2, 2] # Example concentrations # Efficiency of PCR amplification efficiency = 1.8 # Example efficiency # Number of PCR cycles cycles = 30 # Example number of cycles # Simulate PCR concentrations = multiplex_pcr(initial_concentrations, priss # Plot results plt.figure(figure=(10, 6))</pre>	τιοns :
<pre>initial_concentrations = [100, 50, 75] # Example concentra # Primer concentrations primer_concentrations = [2, 2, 2] # Example concentrations # Efficiency of PCR amplification efficiency = 1.8 # Example efficiency # Number of PCR cycles # Simulate PCR concentrations = multiplex_pcr(initial_concentrations, pris # Plot results pl.figure(figure=(10, 6)) for 1, cone in enumerate(concentrations.T):</pre>	tions mer_concentrations, effici
<pre>initial_concentrations = [100, 50, 75] # Example concentra # Primer concentrations primer_concentrations = [2, 2, 2] # Example concentrations # Efficiency of PCR amplification efficiency = 1.8 # Example efficiency # Number of PCR cycles cycles = 30 # Example number of cycles # Simulate PCR concentrations = multiplex_pcr(initial_concentrations, priss # Plot results plt.figure(figure=(10, 6))</pre>	tions mer_concentrations, effici
<pre>initial_concentrations = [100, 50, 75] # Example concentra # Primor concentrations primor_concentrations primor_concentrations # Efficiency of PCR amplification efficiency = 1.8 # Example efficiency # Number of PCR cycles cycles = 30 # Example number of cycles # Simulate PCR concentrations = multiplex_pcr(initial_concentrations, primore # Plot results plt.figure(figsize=(10, 6)) for 1, conc 1 n enumerate(concentrations.T): plt.plot(range(cycles+i), conc, label=f'Template (1+1)' plt.xlabel('Cycle number')</pre>	tions wr_concentrations, effici
<pre>initial_concentrations = [100, 50, 75] # Example concentra # Primer concentrations primer_concentrations = [2, 2, 2] # Example concentrations # Efficiency of PCR amplification efficiency = 1.0 # Example efficiency # Number of PCR cycles cycles = 30 # Example number of cycles # Simulate PCR concentrations = multiplex_pcr(initial_concentrations, pris plutite PCR plt.figure(figure=(10, 6)) for i, cone in enumerate(concentrations.T): plt.plot(frame(cycles+1), cone, label=f'Template (i+i)' plt.xlabel('Cycle number') plt.ylabel('Concentration')</pre>	tions mer_concentrations, effici
<pre>initial_concentrations = [100, 50, 75] # Example concentra # Primer concentrations primer_concentrations = [2, 2, 2] # Example concentrations # Efficiency of PCR amplification efficiency = 1.8 # Example efficiency # Number of PCR cycles cycles = 30 # Example number of cycles # Simulate PCR concentrations = multiplex_pcr(initial_concentrations, prim # Plot results plt.figure(figize=(10, 6)) for 1, conc in enumerate(concentrations.7): plt.plot(range(cycles+i), conc, label=f'Template (i+i)' plt.klabel('Cycle number') plt.klabel('Cycle number') plt.klabel('Concentration')</pre>	tions mer_concentrations, effici
<pre># Initial concentrations of DNA templates (e.g., different initial_concentrations = [100, 50, 75] # Example concentra m Primer_concentrations primer_concentrations = [2, 2, 2] # Example concentrations # Efficiency of PCR amplification efficiency = 1.8 # Example efficiency # Number of PCR cyclos cycles = 30 # Example number of cycles # Simulate PCR concentrations = multiplex_pcr(initial_concentrations, prim # Plot results plt.figure(figsize(10, 6)) for 1, conc in enumerate(concentrations.T): plt.plt(ingne(cycles+1), conc, label=('Template (i+i)' plt.tabel('Concentration') plt.tabel('Concentration') plt.tabel('Concentration') plt.tabel('Dentration') plt.tabel() plt.tabel()</pre>	tions mer_concentrations, effici

Computational Simulation of Multiplex

Amplification Dynamics: This illustrates the simulated dynamics of multiplex PCR amplification, showing changes in DNA template concentrations across multiple cycles. Initial concentrations of DNA templates and primer concentrations are provided as inputs to the simulation, along with the efficiency of PCR amplification and the number of cycles. Each curve represents the amplification profile of a specific DNA template over successive cycles, demonstrating the exponential increase in concentration characteristic of PCR ($\underline{3}$). This computational model offers insights into the behavior of multiplex PCR reactions, essential for optimizing experimental protocols and understanding the dynamics of DNA amplification in complex biological samples ($\underline{1}$).

<pre>import numpy as np # Define probe sequences and target sequences probes = ["ATCG", "GCTA",] # Example probe sequences targets = ["TACG", "GCAT",] # Example target sequences # Simulate hybridization def simulate_hybridization(probes, targets, hybridization_rate): hybridization_matrix = np.zeros((len(probes), len(targets))) for i, probe in enumerate(probes): for j, target in enumerate(targets): # Simulate hybridization_prob = similarity = sum(1 for a, b in zip(probe, target) if a == b) hybridization_matrix[][j] = hybridization_prob * hybridization_rate return hybridization_matrix # Example parameters hybridization_matrix # Example parameters hybridization_matrix = simulate_hybridization(probes, targets, hybridization_rate) print("Hybridization Matrix:") print("Hybridization_matrix)</pre>	python D Copy code
<pre>probes = ["ATCG", "GCTA",] # Example probe sequences targets = ["TACG", "CGAT",] # Example target sequences # Simulate hybridization def simulate hybridization(probes, targets, hybridization_rate): hybridization_matrix = np.zeros((len(probes), len(targets))) for i, probe in enumerate(probes): for j, target in enumerate(targets):</pre>	import numpy as np
<pre>targets = ["TACG", "CGAT",] # Example target sequences # Simulate hybridization def simulate_hybridization(probes, targets, hybridization_rate): hybridization_matrix = np.zeros((len(probes), len(targets))) for i, probe in enumerate(probes): for j, target in enumerate(targets): # Simulate hybridization based on sequence similarity and hybridization n similarity = sum(1 for a, b in zip(probe, target) if a == b) hybridization_prob = similarity / len(probe) hybridization_matrix[i][j] = hybridization_prob * hybridization_rate return hybridization_matrix # Example parameters hybridization_matrix = simulate_hybridization(probes, targets, hybridization_rate) print("Hybridization Matrix:")</pre>	# Define probe sequences and target sequences
<pre># Simulate hybridization def simulate_hybridization(probes, targets, hybridization_rate): hybridization_matrix = np.zeros((len(probes), len(targets))) for i, probe in enumerate(probes): for j, target in enumerate(targets):</pre>	<pre>probes = ["ATCG", "GCTA",] # Example probe sequences</pre>
<pre>def simulate_hybridization(probes, targets, hybridization_rate): hybridization_matrix = np.zeros((len(probes), len(targets))) for i, probe in enumerate(probes): for j, target in enumerate(targets):</pre>	<pre>targets = ["TACG", "CGAT",] # Example target sequences</pre>
<pre>hybridization_matrix = np.zeros((len(probes), len(targets))) for i, probe in enumerate(probes): for j, target in enumerate(targets): # Simulate hybridization based on sequence similarity and hybridization n similarity = sum(1 for a, b in zip(probe, target) if a == b) hybridization_prob = similarity / len(probe) hybridization_matrix[i][j] = hybridization_prob * hybridization_rate return hybridization_matrix # Example parameters hybridization_matrix = simulate_hybridization(probes, targets, hybridization_rate) print("Hybridization Watrix:")</pre>	# Simulate hybridization
<pre>for i, probe in enumerate(probes): for j, target in enumerate(targets): # Simulate hybridization based on sequence similarity and hybridization i similarity = sum(1 for a, b in zip(probe, target) if a == b) hybridization_prob = similarity / len(probe) hybridization_matrix[i][j] = hybridization_prob * hybridization_rate return hybridization_matrix # Example parameters hybridization_rate = 0.8 # Perform simulation hybridization_matrix = simulate_hybridization(probes, targets, hybridization_rate) print("Hybridization Watrix:")</pre>	<pre>def simulate_hybridization(probes, targets, hybridization_rate):</pre>
<pre>for j, target in enumerate(targets): # Simulate hybridization based on sequence similarity and hybridization i similarity = sum(1 for a, b in zip(probe, target) if a == b) hybridization_prob = similarity / len(probe) hybridization_matrix[1][j] = hybridization_prob * hybridization_rate return hybridization_matrix # Example parameters hybridization_rate = 0.8 # Perform simulation hybridization_matrix = simulate_hybridization(probes, targets, hybridization_rate) print("Hybridization Watrix:")</pre>	<pre>hybridization_matrix = np.zeros((len(probes), len(targets)))</pre>
<pre># Simulate hybridization based on sequence similarity and hybridization : similarity = sum(1 for a, b in zip(probe, target) if a == b) hybridization_prob = similarity / len(probe) hybridization_matrix[i][j] = hybridization_prob * hybridization_rate return hybridization_matrix # Example parameters hybridization_rate = 0.8 # Perform simulation hybridization_matrix = simulate_hybridization(probes, targets, hybridization_rate) print("Hybridization Watrix:")</pre>	<pre>for i, probe in enumerate(probes):</pre>
<pre>similarity = sum(1 for a, b in zip(probe, target) if a == b) hybridization_prob = similarity / len(probe) hybridization_matrix[i][j] = hybridization_prob * hybridization_rate return hybridization_matrix # Example parameters hybridization_rate = 0.8 # Perform simulation hybridization_matrix = simulate_hybridization(probes, targets, hybridization_rate) print("Hybridization Watrix:")</pre>	<pre>for j, target in enumerate(targets):</pre>
<pre>hybridization_prob = similarity / len(probe) hybridization_matrix[i][j] = hybridization_prob * hybridization_rate return hybridization_matrix # Example parameters hybridization_rate = 0.8 # Perform simulation hybridization_matrix = simulate_hybridization(probes, targets, hybridization_rate) print("Hybridization Watrix:")</pre>	# Simulate hybridization based on sequence similarity and hybridization 1
<pre>hybridization_matrix[i][j] = hybridization_prob * hybridization_rate return hybridization_matrix # Example parameters hybridization_rate = 0.8 # Perform simulation hybridization_matrix = simulate_hybridization(probes, targets, hybridization_rate) print("Hybridization Watrix:")</pre>	<pre>similarity = sum(1 for a, b in zip(probe, target) if a == b)</pre>
<pre>return hybridization_matrix # Example parameters hybridization_rate = 0.8 # Perform simulation hybridization_matrix = simulate_hybridization(probes, targets, hybridization_rate) print("Hybridization Watrix:")</pre>	<pre>hybridization_prob = similarity / len(probe)</pre>
<pre># Example parameters hybridization_rate = 0.8 # Perform simulation hybridization_matrix = simulate_hybridization(probes, targets, hybridization_rate) print("Hybridization Matrix:")</pre>	hybridization_matrix[i][j] = hybridization_prob * hybridization_rate
<pre>hybridization_rate = 0.8 # Perform simulation hybridization_matrix = simulate_hybridization(probes, targets, hybridization_rate) print("Hybridization Matrix:")</pre>	return hybridization_matrix
<pre># Perform simulation hybridization_matrix = simulate_hybridization(probes, targets, hybridization_rate) print("Hybridization Matrix:")</pre>	# Example parameters
<pre>hybridization_matrix = simulate_hybridization(probes, targets, hybridization_rate) print("Hybridization Matrix:")</pre>	hybridization_rate = 0.8
<pre>print("Hybridization Matrix:")</pre>	# Perform simulation
	hybridization_matrix = simulate_hybridization(probes, targets, hybridization_rate)
<pre>print(hybridization_matrix)</pre>	<pre>print("Hybridization Matrix:")</pre>
	<pre>print(hybridization_matrix)</pre>
Figure 4:	Figure 4:

Computational Simulation of Probe-Target Hybridization: This depicts the hybridization matrix resulting from a computational simulation of the dynamic interactions between probe and target nucleic acid sequences. Each element in the matrix denotes the calculated probability of successful hybridization between a specific probe sequence and its corresponding target sequence. The simulation incorporates factors such as sequence similarity and a predefined hybridization rate, providing quantitative insights into the propensity for probe-target binding. Such computational models serve as invaluable tools for understanding the molecular dynamics underlying nucleic acid interactions, essential for applications spanning molecular diagnostics, genomics, and biotechnology (<u>23</u>). Clinical Implications and Applications

Predictive models tailored for the BioFire FilmArray System hold substantial clinical implications, offering avenues for optimizing patient care (2). Through precise forecasts of system performance under specific conditions, clinicians can enhance diagnostic decision-making and refine patient management strategies (24). These models

[Citation Adeel, B., Nazir, Q., Naveed, U., Naz, A., Abrar, Z. (2024). Advanced predictive modeling for the biofire filmarray system in infectious disease diagnostics. *Biol. Clin. Sci. Res. J.*, **2024**: *1237*. doi: <u>https://doi.org/10.54112/bcsrj.v2024i1.1237</u>]

PCR

enable clinicians to select the most suitable tests based on factors such as sample type, pathogen load, and other pertinent variables, thereby mitigating the necessity for serial testing and minimizing the likelihood of false-negative results (9).

(25) scrutinizing the clinical impact of the BioFire FilmArray System in diagnosing lower respiratory tract infections (LRTIs) among hospitalized patients revealed notable outcomes. Implementation of the system led to a marked reduction in the time required to initiate appropriate antimicrobial therapy and a consequent decrease in hospital stay duration. Similarly, another study assessing the clinical ramifications of the system in diagnosing bloodstream infections reported significant reductions in the time to appropriate antimicrobial therapy and mortality rates.

Predictive models further aid clinicians in identifying patients at heightened risk of developing complications or adverse outcomes. For instance, a model accurately predicting the system's performance in detecting antibiotic resistance genes enables clinicians to identify patients prone to antibiotic-resistant infections, thus facilitating tailored treatment adjustments (25).

Implications for Diagnostic Decision-Making and Patient Management

Predictive models tailored for the BioFire FilmArray System fundamentally alter diagnostic decision-making and patient management paradigms. By accurately forecasting system performance, clinicians can make informed choices regarding test selection, result interpretation, and treatment plan adjustments (24).

If a predictive model indicates diminished sensitivity of the BioFire FilmArray System in detecting specific pathogens within a particular sample type, clinicians may opt for supplementary tests or adapt treatment plans accordingly (26). Conversely, if the model suggests high sensitivity in detecting antibiotic resistance genes, clinicians can modify treatment plans to avoid ineffective antibiotics.

In essence, predictive models for the BioFire FilmArray System serve as indispensable tools in clinical decisionmaking, enabling clinicians to navigate diagnostic complexities with precision and efficacy, ultimately enhancing patient outcomes and healthcare delivery (<u>10</u>). Remaining Challenges and Limitations

Despite progress made in developing predictive models for the BioFire FilmArray System, several challenges remain: Limited availability of large, representative datasets (Existing datasets may lack sufficient diversity in terms of sample types, pathogen loads, and geographic locations, limiting the generalizability of predictive models) (9). Insufficient representation of rare pathogens (Many predictive models focus primarily on common pathogens, neglecting rare ones. As a result, predictive models may fail to capture the full spectrum of pathogens encountered in clinical practice) (27). Ignoring temporal dynamics (Most predictive models assume static conditions, ignoring temporal changes in pathogen prevalence, drug susceptibility patterns, and host immune responses. Accounting for temporal dynamics would greatly improve the accuracy of predictive models) (12). Disregarding individual variability (Predictive models often ignore individual differences in host genetics, immunity, and comorbidities, which can significantly impact the performance of the BioFire FilmArray System. Including individual variability in predictive models would improve their accuracy and applicability) (<u>28</u>). Failure to integrate multi-omics data (Integrating multi-omics data, such as genomics, transcriptomics, proteomics, and metabolomics, could potentially improve the accuracy of predictive models) (<u>13</u>). However, this remains a challenging task due to the complex nature of multi-omics data integration.

Future research directions aimed at overcoming the identified challenges

Collecting larger, more diverse datasets will enable the development of more robust and generalizable predictive models. Collaborative efforts between academic institutions, hospitals, and commercial entities could facilitate the creation of shared databases.

Investigators should prioritize the inclusion of rare pathogens in predictive models to ensure that these models reflect the complete spectrum of pathogens encountered in clinical practice. Developing dynamic predictive models that account for temporal changes in pathogen prevalence, drug susceptibility patterns, and host immune responses would greatly improve the accuracy of predictive models.

Incorporating individual differences in host genetics, immunity, and comorbidities into predictive models would improve their accuracy and applicability.

Combining multi-omics data sources could potentially improve the accuracy of predictive models. However, this remains a challenging task due to the complex nature of multi-omics data integration. Investigators should explore novel computational tools and algorithms designed to handle multi-omics data integration.

Validating predictive models in real-time clinical settings would provide valuable insights into their practical application and inform necessary modifications.

Conclusion

The BioFire FilmArray System is a powerful tool in clinical microbiology, capable of detecting multiple pathogens simultaneously. However, the accuracy and sensitivity of the system may be influenced by various factors, including sample types, pathogen load, storage conditions, and assay design characteristics. Predictive models can help optimize the performance of the BioFire FilmArray System under varying conditions, guiding clinicians toward choosing the most suitable tests based on sample type, pathogen load, and other relevant factors. Future research directions aimed at overcoming the identified challenges include expanding dataset size and diversity, incorporating rare pathogens, considering temporal dynamics, addressing individual variability, integrating multi-omics data, and prospectively validating model predictions in clinical settings. In clinical practice, predictive models for the BioFire FilmArray System can have significant implications for optimizing patient care, helping clinicians make informed diagnostic decisions, and improving patient management. By accurately predicting the performance of the system under specific conditions, clinicians can adjust treatment plans accordingly, reducing the need for serial testing and minimizing the risk of false-negative results. In research, predictive modeling has the potential to enhance the accuracy and sensitivity of the BioFire FilmArray System in clinical microbiology, ultimately

improving patient care and public health outcomes. By addressing the identified challenges and exploring future research directions, investigators can develop more accurate and useful predictive models for the BioFire FilmArray System.

Declarations

Data Availability statement

All data generated or analyzed during the study are included in the manuscript.

Ethics approval and consent to participate Approved by the department Concerned. Consent for publication

Approved **Funding** Not applicable

Conflict of interest

The authors declared absence of conflict of interest.

Author Contribution

BUSHRA ADEEL

Coordination of collaborative efforts. Study Design, Review of Literature.

QURATULAIN NAZIR

Conception of Study, Development of Research Methodology Design, Study Design, Review of manuscript, final approval of manuscript. Conception of Study, Final approval of manuscript. USHNA NAVEED Manuscript revisions, critical input.

Coordination of collaborative efforts.

AYESHA NAZ

Data acquisition, analysis. Manuscript drafting.

ZUNAIRA ABRAR

Data entry and Data analysis, drafting article. Data acquisition, analysis. Coordination of collaborative efforts.

References

1. Trujillo-Gomez J, Tsokani S, Arango-Ferreira C, Atehortua-Munoz S, Jimenez-Villegas MJ, Serrano-Tabares C, et al. Biofire FilmArray Meningitis/Encephalitis panel for the aetiological diagnosis of central nervous system infections: a systematic review and diagnostic test accuracy meta-analysis. EClinicalMedicine. 2022;44.

2. Tansarli GS, Chapin KC. Diagnostic test accuracy of the BioFire® FilmArray® meningitis/encephalitis panel: a systematic review and meta-analysis. Clinical Microbiology and Infection. 2020;26(3):281-90.

3. Sze DT, Lau CC, Chan T-M, Ma ES, Tang BS. Comparison of novel rapid diagnostic of blood culture identification and antimicrobial susceptibility testing by Accelerate Pheno system and BioFire FilmArray Blood Culture Identification and BioFire FilmArray Blood Culture Identification 2 panels. BMC microbiology. 2021;21:1-15. 4. Leber AL, Everhart K, Daly JA, Hopper A, Harrington A, Schreckenberger P, et al. Multicenter evaluation of BioFire FilmArray respiratory panel 2 for detection of viruses and bacteria in nasopharyngeal swab samples. Journal of Clinical Microbiology. 2018;56(6):10.1128/jcm. 01945-17.

5. Fhooblall M, Nkwanyana F, Mlisana KP. Evaluation of the BioFire® FilmArray® Blood Culture Identification Panel on positive blood cultures in a regional hospital laboratory in KwaZulu-Natal. African Journal of Laboratory Medicine. 2016;5(1):1-8.

6. Jones S, Janto C, Balgahom R, Samarasekara H, Branley J. Evaluation of the biofire filmarray blood culture identification panel on positive blood cultures in a regional hospital laboratory. Pathology. 2018;50:S114-S5.

7. Cortazzo V, D'Inzeo T, Giordano L, Menchinelli G, Liotti FM, Fiori B, et al. Comparing BioFire FilmArray BCID2 and BCID panels for direct detection of bacterial pathogens and antimicrobial resistance genes from positive blood cultures. Journal of Clinical Microbiology. 2021;59(4):10.1128/jcm. 03163-20.

8. Ward C, Stocker K, Begum J, Wade P, Ebrahimsa U, Goldenberg S. Performance evaluation of the Verigene®(Nanosphere) and FilmArray®(BioFire®) molecular assays for identification of causative organisms in bacterial bloodstream infections. European Journal of Clinical Microbiology & Infectious Diseases. 2015;34:487-96.

9. Machiels JD, Cremers AJ, van Bergen-Verkuyten MC, Paardekoper-Strijbosch SJ, Frijns KC, Wertheim HF, et al. Impact of the BioFire FilmArray gastrointestinal panel on patient care and infection control. PLoS One. 2020;15(2):e0228596.

10. Edin A, Eilers H, Allard A. Evaluation of the Biofire Filmarray Pneumonia panel plus for lower respiratory tract infections. Infectious Diseases. 2020;52(7):479-88.

11. Southern TR, VanSchooneveld TC, Bannister DL, Brown TL, Crismon AS, Buss SN, et al. Implementation and performance of the BioFire FilmArray® Blood Culture Identification panel with antimicrobial treatment recommendations for bloodstream infections at a midwestern academic tertiary hospital. Diagnostic Microbiology and Infectious Disease. 2015;81(2):96-101.

12. Kyriazopoulou E, Karageorgos A, Liaskou-Antoniou L, Koufargyris P, Safarika A, Damoraki G, et al. BioFire® FilmArray® Pneumonia Panel for severe lower respiratory tract infections: subgroup analysis of a randomized clinical trial. Infectious Diseases and Therapy. 2021;10:1437-49.

13. Murphy CN, Fowler R, Balada-Llasat JM, Carroll A, Stone H, Akerele O, et al. Multicenter evaluation of the BioFire FilmArray pneumonia/pneumonia plus panel for detection and quantification of agents of lower respiratory tract infection. Journal of Clinical Microbiology. 2020;58(7):10.1128/jcm. 00128-20.

14. Obaro S, Hassan-Hanga F, Medugu N, Olaosebikan R, Olanipekun G, Jibir B, et al. Comparison of bacterial culture with BioFire® FilmArray® multiplex PCR screening of archived cerebrospinal fluid specimens from children with suspected bacterial meningitis in Nigeria. BMC Infectious Diseases. 2023;23(1):641.

15. Gastli N, Loubinoux J, Daragon M, Lavigne J-P, Saint-Sardos P, Pailhoriès H, et al. Multicentric evaluation

of BioFire FilmArray Pneumonia Panel for rapid bacteriological documentation of pneumonia. Clinical Microbiology and Infection. 2021;27(9):1308-14.

16. Kosai K, Akamatsu N, Ota K, Mitsumoto-Kaseida F, Sakamoto K, Hasegawa H, et al. BioFire FilmArray Pneumonia Panel enhances detection of pathogens and antimicrobial resistance in lower respiratory tract specimens. Annals of Clinical Microbiology and Antimicrobials. 2022;21(1):24.

17. Mponponsuo K, Leal J, Spackman E, Somayaji R, Gregson D, Rennert-May E. Mathematical model of the cost-effectiveness of the BioFire FilmArray Blood Culture Identification (BCID) Panel molecular rapid diagnostic test compared with conventional methods for identification of Escherichia coli bloodstream infections. Journal of Antimicrobial Chemotherapy. 2022;77(2):507-16.

18. Acuña M, Benadof D, Yohannessen K, Leiva Y, Clement P. FilmArray® Meningoencephalitis panel in the diagnosis of central nervous system infections: stewardship and cost analysis in a paediatric hospital in Chile. BMC pediatrics. 2022;22(1):1-11.

19. Mponponsuo K. BioFire FilmArray Blood Culture system cost saving for identifying E. coli bloodstream infections. PharmacoEconomics & Outcomes News. 2021;892:4-27.

20. Ng AH, Hu H, Wang K, Scherler K, Warren SE, Zollinger DR, et al. Organ-specific immunity: A tissue analysis framework for investigating local immune responses to SARS-CoV-2. Cell Reports. 2023;42(10).

21. Doyle B, Rihal CS, O'Sullivan CJ, Lennon RJ, Wiste HJ, Bell M, et al. Outcomes of stent thrombosis and restenosis during extended follow-up of patients treated with bare-metal coronary stents. 2007;116(21):2391-8.

22. Murphy C, Fowler R, Iwen PC, Fey PD. Evaluation of the BioFire FilmArray® GastrointestinalPanel in a Midwestern academic hospital. European Journal of Clinical Microbiology & Infectious Diseases. 2017;36:747-54.

23. Berinson B, Both A, Berneking L, Christner M, Lütgehetmann M, Aepfelbacher M, et al. Usefulness of BioFire FilmArray BCID2 for blood culture processing in clinical practice. Journal of Clinical Microbiology. 2021;59(8):10.1128/jcm. 00543-21.

24. Buchan BW, Windham S, Balada-Llasat J-M, Leber A, Harrington A, Relich R, et al. Practical comparison of the BioFire FilmArray pneumonia panel to routine diagnostic methods and potential impact on antimicrobial stewardship in adult hospitalized patients with lower respiratory tract infections. Journal of Clinical Microbiology. 2020;58(7):10.1128/jcm. 00135-20.

25. Markovich K, Wingler MJB, Stover KR, Barber KE, Wagner JL, Cretella DA. Analysis of the Clinical Impact of the BioFire FilmArray Meningitis Encephalitis Panel on Antimicrobial Use and Duration of Therapy at an Academic Medical Center. Diseases. 2022;10(4):110.

26. Carmon D, Rohana H, Azrad M, Peretz A. The Impact of a Positive Biofire® FilmArray® Gastrointestinal Panel Result on Clinical Management and Outcomes. Diagnostics. 2023;13(6):1094.

27. Alashkar G, Parslow G. Evaluation of the BioFire Filmarray-2.0 Meningitis/Encephalitis and blood culture identification panels. New Zealand Journal of Medical Laboratory Science. 2020;74(3):228-9.

28. Rule R, Paruk F, Becker P, Neuhoff M, Chausse J, Said M. Clinical utility of the BioFire FilmArray Blood Culture Identification panel in the adjustment of empiric antimicrobial therapy in the critically ill septic patient. PLoS One. 2021;16(7):e0254389.



Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit http://creativecommons.org/licen ses/by/4.0/. © The Author(s) 2024

[[]Citation Adeel, B., Nazir, Q., Naveed, U., Naz, A., Abrar, Z. (2024). Advanced predictive modeling for the biofire filmarray system in infectious disease diagnostics. *Biol. Clin. Sci. Res. J.*, **2024**: *1237*. doi: <u>https://doi.org/10.54112/bcsrj.v2024i1.1237</u>]