

**Julius-Maximilians-Universität Würzburg**

Studiengang: Master Biowissenschaften (4 Semester, 120 ECTS)

# **Development of an Image Analysis and Modelling Framework for Influenza Genome Assembly**

## **Master Thesis**

Eingereicht von Jannis Witte



Center for Computational and Theoretical Biology  
BioMedical Data Science Group

Erstbetreuer:	Dr. Markus Ankenbrand
Zweitbetreuerin:	Prof. Dr. Sabine Fischer
Abgabedatum:	14.07.2025

# Inhaltsverzeichnis

Zusammenfassung .....	4
1. Abkürzungen .....	6
2. Einleitung .....	7
2.1 Aufbau des Genoms und Replikationsmechanismus von Influenzaviren .....	7
2.2 In-situ Sequenzierung .....	9
2.3 Zielsetzung .....	11
3. Material und Methoden .....	12
3.1 Software und Code-Verfügbarkeit .....	12
3.2 Datensatz .....	12
3.3 Manuelle Annotation .....	14
3.4 Analysepipeline .....	15
3.4.1 Registrierung .....	16
3.4.2 Hintergrundfilterung .....	17
3.4.3 Normalisierung .....	18
3.4.4 Maskierung signalintensiver Regionen .....	19
3.4.5 Spot-basierte Dekodierung .....	22
3.4.6 Pixel-basierte Dekodierung .....	22
3.4.7 Zellkernsegmentierung .....	24
3.5 Berechnung von Komplexwahrscheinlichkeiten .....	24
3.6 Modellierung der Genomassemblierung .....	24
4. Ergebnisse .....	25
4.1 Evaluation der Analysepipeline .....	25
4.1.1 Modellbasierte Maskierung .....	25
4.1.2 Spot-basierte Dekodierung .....	26
4.1.3 Pixel-basierte Dekodierung .....	33
4.2 Analyse der Genomassemblierung .....	43

4.2.1	Relative Anzahlen der Komplexe.....	43
4.2.2	Abundanz und Wahrscheinlichkeiten der Komplexe .....	48
4.2.3	Modellierung einer monomeren Assemblierung.....	54
5.	Diskussion.....	55
5.1	Pipeline Qualität und Verbesserungen.....	55
5.2	Möglicher Mechanismus der Genomassemblierung .....	59
5.3	Ausblick.....	61
6.	Referenzen.....	63
7.	Danksagung.....	66
8.	Anhang.....	67
	Anhang 1 .....	67
	Anhang 2 .....	75

# Zusammenfassung

## Deutsch

In dieser Arbeit wurde die Assemblierung des Influenzavirusgenoms mithilfe von In-situ Sequenzierung und multiplexem, direktem „Padlock-Probing“ untersucht. Ziel war es, die Zusammensetzung von einzelnen viralen Ribonukleoproteinkomplexen (vRNPs) zu größeren Komplexen zu detektieren und daraus Rückschlüsse auf die Mechanismen der Genomassemblierung zu ziehen. Ein zentraler Bestandteil der Arbeit war die Entwicklung einer automatisierten Analysepipeline, die eine zuverlässige Detektion der einzelnen vRNP-Segmente sowie deren Zusammenschluss zu größeren Komplexen innerhalb der Fluoreszenzmikroskopiebilder ermöglichte. Die Pipeline umfasste sämtliche Schritte der Präprozessierung. Darunter die Registrierung der Bilder über Sequenzierungen hinweg, eine Hintergrundfilterung zur Entfernung von Rauschen, sowie eine Normalisierung der Intensitätswerte zum Ausgleichen von Unterschieden zwischen den Field of Views (FOVs) und Farbkanälen. Zusätzlich wurde eine Maskierung signalintensiver Regionen integriert, um zwei unterschiedliche Dekodierungsmethoden in dichten und undichten Regionen zu vergleichen. Die erste Methode war die Spot-basierte Dekodierung (SBD), welche zunächst punktförmige Intensitätsmaxima detektierte und anschließend über Sequenzierungen und Kanäle verfolgte. Diese Methode lieferte vor allem in undichten Regionen gute Ergebnisse. Die Pixel-basierte Dekodierung (PBD) beruhte auf der Dekodierung von Einzelpixeln, welche im Anschluss zu Spots gruppiert wurden. Es wurde angenommen, dass die Betrachtung einzelner Pixel vor allem in dichten Regionen, wo sich viele Signale überlagern, einen Vorteil bietet. Die Ergebnisse haben allerdings gezeigt, dass die PBD sowohl in dichten als auch in undichten Regionen fehleranfälliger und inkonsistenter als die SBD war. Daher wurde für die folgende Analyse der vRNP-Komplexe die SBD in undichten Regionen angewandt.

Die Analyse der vRNP-Segmente zeigte, dass bestimmte Segmentkombinationen, insbesondere die Polymerasesegmente (PB2, PB1, PA) sowie M und NS, deutlich häufiger vorkamen als bei zufälliger Assemblierung zu erwarten gewesen wäre. Dies konnte durch den Vergleich von beobachteten Abundanzen und berechneten Komplexwahrscheinlichkeiten gezeigt werden. Die Modellierung einer monomeren Assemblierung bestätigte diese Ergebnisse. Insgesamt zeigte die Analyse, dass kein fester, aber auch kein rein zufälliger Assemblierungsweg vorlag. Die Assemblierung folgte vielmehr spezifischen Interaktionsmustern, die durch RNA-RNA Interaktionen vermittelt wurden und die Bildung bestimmter Subkomplexe begünstigten. Die Ergebnisse stimmten mit bisherigen Studien überein.

## English

In this study, the assembly of the influenza virus genome was investigated using in-situ sequencing and multiple, direct padlock probing. The aim was to detect the assembly of individual viral ribonucleoprotein complexes (vRNPs) into larger complexes and to draw conclusions about the mechanisms of genome assembly. A central component of the work was the development of an automated analysis pipeline that enabled reliable detection of the individual vRNP segments and their assembly into larger complexes within the fluorescence microscopy images. The pipeline included all preprocessing steps. This included registration of images across sequencing rounds, background filtering to remove noise, and normalization of intensity values to compensate for differences between field of views (FOVs) and color channels. In addition, masking of signal-intensive regions was integrated to enable comparison of two different decoding methods in dense and non-dense regions. The first method was spot-based decoding (SBD), which first detected point intensity maxima and then tracked them across sequencing rounds and channels. This method delivered good results, particularly in non-dense regions. Pixel-based decoding (PBD) was based on the decoding of individual pixels, which were then grouped into spots. It was assumed that looking at individual pixels offers an advantage, especially in dense regions where many signals overlap. However, the results showed that PBD was more error-prone and inconsistent than SBD in both dense and non-dense regions. Therefore, SBD was applied in non-dense regions for the following analysis of vRNP complexes.

The analysis of the vRNP segments showed that certain segment combinations, in particular the polymerase segments (PB2, PB1, PA) as well as M and NS, occurred significantly more frequently than expected under random assembly. This was demonstrated by comparing observed abundances and calculated complex probabilities. Modeling of a monomeric assembly pathway confirmed these results. Overall, the analysis showed that there is no fixed, but also no purely random assembly pathway. Rather, assembly follows specific interaction patterns that are mediated by RNA-RNA interactions and favor the formation of certain subcomplexes. The results were consistent with findings from previous studies.

# 1. Abkürzungen

DAPI	4',6-Diamidino-2-phenylindol
FOV	Field of View
hpi	hours post infection
ICC	Instant Computational Clearing
ISS	In-situ Sequenzierung
MIP	Maximale Intensitätsprojektion
PBD	Pixel-basierte Dekodierung
RCP	Rolling Circle Product
SBD	Spot-basierte Dekodierung
vRNP	Virales Ribonukleoprotein

## 2. Einleitung

Das Influenzavirus gehört zu den Orthomyxoviren und ist als Erreger der saisonalen Grippe weltweit von großer medizinischer Bedeutung. Jährlich verursacht es weltweit etwa drei bis fünf Millionen schwere Erkrankungsfälle und bis zu 650.000 Todesfälle (WHO 2025). Das Virus besitzt ein segmentiertes Genom, das aus insgesamt acht einzelnen RNA-Segmenten besteht. Diese Segmentierung ermöglicht dem Virus, durch sogenannte genetische Reassortierung schnell neue Varianten zu erzeugen, die für die hohe Variabilität und pandemische Ausbreitung verantwortlich sind. Die korrekte Assemblierung der acht RNA-Segmente zu einem vollständigen Virion ist ein komplex regulierter und bisher nicht vollständig verstandener Mechanismus. Ein besseres Verständnis dieses Prozesses könnte zur Entwicklung neuer antiviraler Medikamente beitragen, indem gezielt in die Verpackung des viralen Genoms eingegriffen wird.

### 2.1 Aufbau des Genoms und Replikationsmechanismus von Influenzaviren

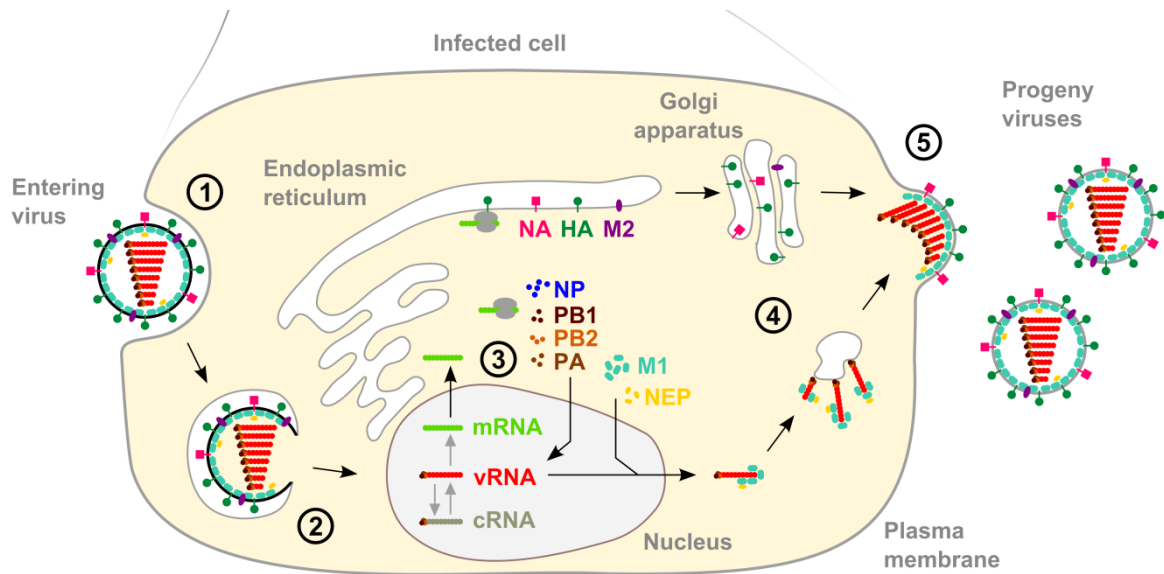
Das Influenzavirus ist ein behülltes Virus mit einer sphärischen Form von etwa 80-120 nm Durchmesser. Die Virushülle besteht hauptsächlich aus den Oberflächenproteinen Hämagglutinin (HA) und Neuramidase (NA). Während das HA-Protein für die Bindung an zelluläre Rezeptoren verantwortlich ist, spielt NA eine entscheidende Rolle bei der Freisetzung neuer Viruspartikel aus der Wirtszelle (Gamblin und Skehel 2010). Das Erbgut des Influenzavirus besteht aus acht einzelsträngigen RNA-Segmenten negativer Polarität, die jeweils mit Nukleoproteinen (NP) und der viralen RNA-Polymerase zu sogenannten viralen Ribonukleoproteinkomplexen (vRNPs) verbunden sind (Eisfeld et al. 2015). Die virale Polymerase besteht aus drei Untereinheiten und wird durch die RNAs PB1, PB2 und PA codiert (Pflug et al. 2014). Zwei weitere RNAs codieren die Oberflächenproteine HA und NA. Die restlichen RNAs codieren das Nukleoprotein (NP), die Matrixproteine (M1, M2) und die Nichtstrukturellen-Proteine (NS1, NS2).

Die Replikation des Influenzavirus innerhalb des Wirts beginnt mit der Bindung des Oberflächenproteins HA an spezifische Rezeptoren auf der Zelloberfläche. Diese enthalten Sialinsäure und ermöglichen die Endozytose des Viruspartikels in die Wirtszelle (Hamilton et al. 2012) (**Abb. 1**, (1)). Innerhalb des Endosoms herrscht ein saures Milieu, was die Fusion der Virusmembran mit der Endosomenmembran induziert. Die vRNPs werden ins Zytoplasma freigesetzt und zum Zellkern transportiert (O'Neill et al. 1995) (2). Im Zellkern werden die viralen RNAs in positive mRNAs transkribiert, die dann durch die zelluläre

Translationsmaschinerie abgelesen werden können. Parallel dazu synthetisiert die virale RNA-Polymerase komplementäre RNAs positiver Polarität., die wiederum als Vorlage zur Synthese neuer genomischer viraler RNA-Segmente negativer Polarität dienen (Te Velthuis und Fodor 2016) (3). Die neu synthetisierten viralen RNA-Segmente werden mit Nukleoproteinen und Polymerasen assoziiert, sodass neue vRNP-Komplexe entstehen. Die vRNPs werden im Anschluss aus dem Zellkern exportiert. Nach dem nukleären Export verbinden sich die vRNPs im Zytoplasma mit Rab11-positiven Recycling-Endosomen, welche über das Mikrotubuli-Netzwerk zur Plasmamembran transportiert werden (Amorim et al. 2011) (4). Dort lagern sich die vRNPs unter Beteiligung des Matrixproteins zu neuen Viruspartikeln zusammen. Abschließend werden diese durch Knospung von der Zellmembran abgeschnürt (5). Dieser Prozess wird durch die virale Neuraminidase erleichtert. Die freigesetzten Viruspartikel können daraufhin verbreitet werden und neue Wirtszellen befallen (Bouvier und Palese 2008).

Die Verpackung der acht vRNP-Segmente an der Wirtsmembran folgt dabei einem selektiven Prozess (Noda und Kawaoka 2010). Experimente haben gezeigt, dass die vRNPs nicht in beliebiger Reihenfolge eingebaut werden, sondern sich entlang spezifischer Verpackungssignale zusammensetzen. Diese Signale werden durch spezifische RNA-RNA-Interaktionen an den 3'- und 5'-Enden der vRNAs vermittelt. Sie ermöglichen eine hochselektive Verpackung, sodass typischerweise genau ein Exemplar jedes vRNP-Segments pro Viruspartikel vorliegt (Noda 2021). Strukturelle Untersuchungen mittels Elektronentomographie haben gezeigt, dass vRNPs in einer charakteristischen „1+7“-Konfiguration vorliegen. Das bedeutet ein zentrales vRNP-Segment liegt im Zentrum des Virus, das von den restlichen sieben Segmenten umgeben ist (Noda et al. 2006). Eine exakte Reihenfolge, mit der die Segmente integriert werden, ist bisher nicht vollständig geklärt. Es wird angenommen, dass die Assemblierung mit der Bildung kleinerer Subkomplexe beginnt, also zwei oder mehreren zusammenhängenden vRNPs, die sich bereits im Zellkern bilden (Bouvier und Palese 2008). Durch das Ausschalten einzelner Verpackungssignale konnte gezeigt werden, dass die Signale einiger Segmente einen größeren Einfluss auf die korrekte Verpackung haben als andere. Während die Signale der PB1-, HA - und NS-Segmente keinen Einfluss auf das Wachstum hatten, sorgte das Ausschalten der Signale bei PB2, PA, NP und M für ein beeinträchtigtes Wachstum (Gao et al. 2012).





**Abb. 1: Schematische Darstellung des Replikationszyklus des Influenzavirus.** (1) Das Virus haftet mittels Hämagglutinin an spezifische Rezeptoren an der Zelloberfläche des Wirts und induziert die Aufnahme ins Zytoplasma mittels Endozytose. (2) Säureinduzierte Fusion der Virusmembran mit der Endosomenmembran und der Freisetzung der vRNPs ins Zytoplasma, gefolgt vom Transport der vRNPs zum Zellkern. (3) Transkription der vRNA zu cRNA und mRNA im Zellkern und Export der mRNAs ins Zytoplasma mit anschließender Translation von NP-, PB1-, PB2- und PA-Proteinen. Anschließend Import der Proteine in den Zellkern, wo sie weitere Transkription fördern. Parallel dazu werden NA, HA, und M2 am Endoplasmatischen Reticulum translatiert. Außerdem Export der vRNPs ins Zytoplasma und Fusionierung mit M1 und NEP. (4) Assoziation der vRNPs mit Rab11-positiven Recycling-Endosomen und Mikrotubuli-vermittelter Transport zur Plasmamembran. Paralleler Transport von NA-, HA- und M2-Proteinen über den Golgi-Apparat zur Plasmamembran. (5) Zusammenlagerung der vRNPs an der Plasmamembran und Abschnürung zu neuen Viruspartikeln. Verändert aus (Max Planck Institute Magdeburg 30.06.25).

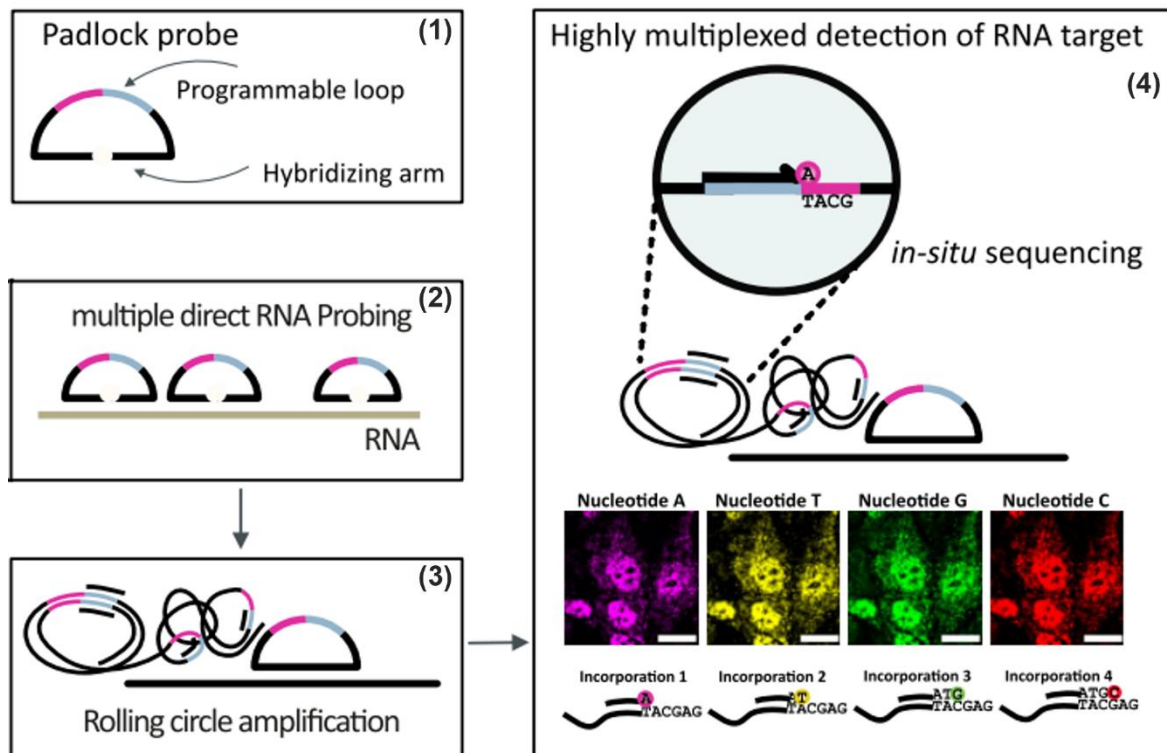
## 2.2 In-situ Sequenzierung

Um den Infektionsprozess von Zellen mit dem Influenzavirus genauer analysieren zu können, wurde ein Verfahren aus In-situ Sequenzierung (ISS), kombiniert mit direktem „Padlock-Probing“ (PLP) verwendet. Die ISS ermöglicht die räumliche Auflösung von RNA-Molekülen in intakten Zellen und ermöglicht so die Analyse von Verteilung und Expression. Dabei werden einzelsträngige DNA-Oligonukleotide (Padlock-Probes) verwendet, deren Enden komplementär an benachbarte Regionen von cDNAs binden. Nach der korrekten Hybridisierung werden die PLPs enzymatisch an die Zielsequenz ligiert und im Anschluss amplifiziert und detektiert (Ke et al. 2013).

In klassischen Ansätzen wird zunächst die RNA in cDNA umgeschrieben, die dann als Ziel für die PLPs dient. Im Gegensatz dazu hybridisieren die PLPs beim multiplen, direkten PLP direkt

an die RNA-Moleküle (**Abb. 2**). Zudem binden gleich mehrere PLPs pro RNA, wodurch die Effizienz der Detektion um das Dreifache steigt (Ahmad et al. 2025).

Während der Durchführung werden die Zellen zunächst infiziert und anschließend fixiert. So bleiben die RNA-Moleküle an ihrer ursprünglichen Position erhalten. Um den Zugang für Enzyme und Sonden zu ermöglichen, werden die Zellmembranen permeabilisiert. Die frei zugängliche RNA dient dann als Ziel für die PLPs. Die PLPs stellen einzelsträngige DNA-Oligonukleotide dar, dessen Enden komplementär zu den benachbarten Bereichen der Zielsequenz sind. In der Mitte befindet sich ein programmierbarer Barcode, der zur Identifizierung der gebundenen RNA genutzt wird. Sobald die Enden der PLPs korrekt hybridisiert haben, werden sie enzymatisch ligiert. Die dabei verwendete PBCV-1 Ligase ist spezifisch und arbeitet nur, wenn beide Enden der Probe korrekt hybridisiert sind (Lohman et al. 2014). Es entsteht eine zirkuläre DNA, welche in einer anschließenden „Rolling Circle Amplifikation“ amplifiziert wird. Das entstandene „Rolling Circle Produkt“ (RCP) enthält somit eine Vielfache Kopie der Zielsequenz. Anschließend wird in der Bildgebung der an das RCP gebundene Barcode ausgelesen. Dabei wird die eindeutige definierte Sequenz über mehrere Zyklen der ISS identifiziert. In jeder Sequenzierrunde wird eine einzelne Base durch ein korrespondierendes fluoreszenzmarkiertes Nukleotid sichtbar gemacht. Die Signale werden in aufeinanderfolgenden Sequenzierrunden mittels Fluoreszenzmikroskopie aufgenommen. Durch die spezifische Abfolge der Fluoreszenzfarben über die Runden lässt sich die zugrundeliegende Sequenz identifizieren, wodurch die zugehörige RNA eindeutig bestimmt werden kann.



**Abb. 2: Schematische Darstellung der In-situ Sequenzierung mit multiplem, direktem „Padlock-Probing“ (PLP).** (1) Die Padlock-Probe besteht aus einem programmierbaren Mittelteil und hybridisierenden Enden, welche komplementär zur Zielsequenz sind. (2) Multiple Bindung und Legierung von PLPs direkt an die RNA und Entstehung einer zirkulären DNA. (3) Amplifikation der PLPs mittels „Rolling Circle Amplifikation“. (4) Detektion der Zielsequenzen über In-situ Sequenzierung der PLPs. Mehrere Runden der Inkorporation von fluoreszent markierten Nukleotiden und Bildgebung ermöglichen die spezifische Detektion der Zielsequenzen. Verändert aus (Ahmad et al. 2025).

## 2.3 Zielsetzung

Ziel dieser Arbeit war die Entwicklung einer umfassenden Analysepipeline zur Auswertung von ISS-Daten aus Influenza-infizierten Zellen, um den Mechanismus der Genomassemblierung besser zu verstehen. Dafür sollte eine Analysepipeline entwickelt werden, die alle Schritte der Präprozessierung, der Detektion der Segmente und der Segmentierung von Zellkompartimenten vereint. Zusätzlich sollte sowohl die Qualität der gesamten Pipeline als auch die Qualität einzelner Schritte validiert werden. Darüber hinaus sollten mögliche Mechanismen der Assemblierung weiter untersucht werden, indem Häufigkeiten verschiedener Einzelsegmente und Segmentkomplexe mit berechneten Wahrscheinlichkeiten verglichen wurden. Weiterhin sollten mithilfe einer Modellierung, in der eine monomere Assemblierung angenommen wird, die wahrscheinlichen Assemblierungswege berechnet

werden. Basierend auf den Ergebnissen der Komplexwahrscheinlichkeiten und der Modellierung sollte dann ein Assemblierungsweg vorgeschlagen werden.

## 3. Material und Methoden

### 3.1 Software und Code-Verfügbarkeit

Die Analysen wurden auf einem Linux-basierten System durchgeführt (Ubuntu 24.04.2). Es wurde die Programmiersprache Python (Version 3.9.21), der Paketmanager Mamba (Version 24.11.1), Git (Version 2.43), sowie die in **Tab. 1** aufgelisteten Python-Pakete verwendet. Der gesamte Code der Analysen ist verfügbar unter: <https://github.com/janniswi00/master>.

**Tab. 1: Für die Analysen verwendeten Python-Pakete mit Anwendung und Referenz.**

<b>Paket</b>	<b>Anwendung</b>	<b>Referenz</b>
<i>glob</i>	Verwaltung von Dateipfaden	(McKinney 2010; Elsdoerfer 2011)
<i>itertools</i>	Ermittlung von Kombinationsmöglichkeiten	(van Rossum 2020b)
<i>matplotlib</i> (3.7.5)	Graphische Darstellung von Bildern	(Hunter 2007)
<i>numpy</i> (1.26.4)	Statistische Berechnungen	(Harris C. R. et al. 2020)
<i>pandas</i> (2.2.3)	Datenmanipulation und Speicherung	(McKinney 2010)
<i>scikit-image</i> (0.24.0)	Einlesen von Bildern	(Van der Walt et. al. 2014)
<i>scikit-learn</i> (1.6.1)	Regressionsmodell und Gruppierung von Pixeln	(Pedregosa et al. 2012)
<i>scipy</i> (1.13.1)	Statistische Berechnungen und Bildverarbeitung	(Virtanen et al. 2020)
<i>starfish</i> (0.3.2)	Auswertung von In-situ Sequenzierungsdaten	(Axelrod et al. 2018)

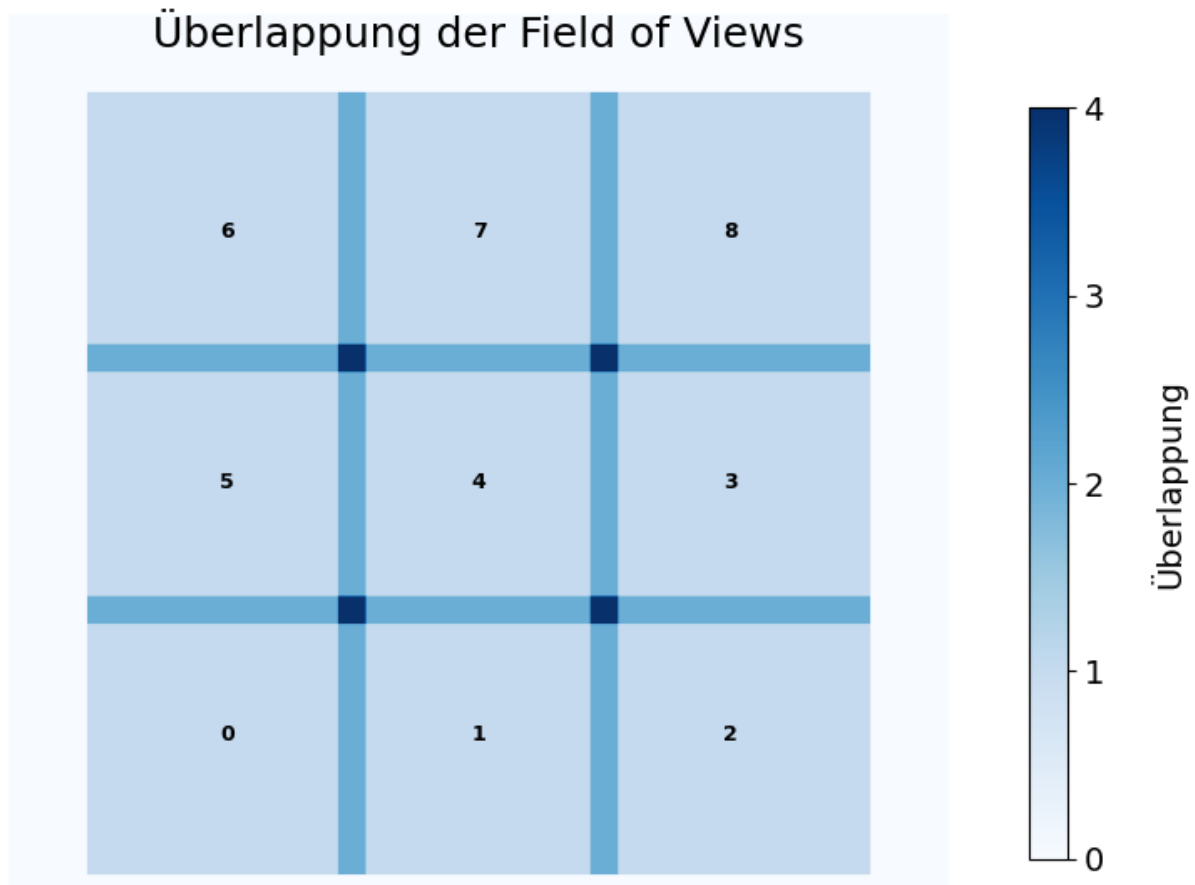
### 3.2 Datensatz

Der Datensatz besteht aus Bildern von MDCK-Zellen, die mit dem Influenza H1N1-Stamm A/Puerto Rico/8/34 (PR8) infiziert wurden. Zur Visualisierung der viralen Transkription wurden die acht viralen Ribonukleoproteinkomplexe (vRNPs) mit multiplen Padlock Probes (PLPs) direkt markiert und in einer In-situ Sequenzierung (ISS) (**Abb. 2**) aufgenommen. Alle Bildaufnahmen in diesem Projekt wurden von dem Helmholtz-Institut für RNA-basierte Infektionsforschung in Würzburg durchgeführt. Die Bildaufnahme wurden mit einem Epifluoreszenzmikroskop von Leica, das über eine automatisierte XYZ-Stufensteuerung sowie über einen Hardware Autofokus verfügte. Für die Fluoreszenzanregung wurde eine LED-Lichtquelle verwendet. Die Bilder wurden mit einer Leica-DFC9000GT-VSC11903-Kamera

aufgenommen. Die Fluoreszenzmarkierung der detektierten Basen erfolgte mit FluorT-markierten Primern von Integrated DNA Technologies. Der Infektionsprozess wurde stündlich von null bis acht Stunden nach Infektion (hours post infection, hpi) aufgezeichnet. Die Aufnahme erfolgte abschnittsweise in neun bis zwölf Sichtfeldern (field of views, FOVs), die sich in den Randbereichen überlagerten. Insgesamt wurde die Analyse für fünf Replikate durchgeführt. Eine beispielhafte Anordnung der FOVs aus Replikat null, null Stunden nach Infektion ist in **Abb. 3** dargestellt. Ein FOV umfasste mehrere Kanäle, darunter einen DAPI-Kanal zur Darstellung der Zellkerne, sowie vier farbliche Fluoreszenzbilder, die den vier DNA-Basen entsprachen. Die Barcodes der PLPs hatten eine Länge von drei Basen, sodass die Sequenzierung über drei Runden erfolgte. Die Barcodes der PLPs, die zur Markierung der vRNPs verwendet wurden sind in **Tab. 2** gelistet. Während der Aufnahme wurden 23 Bilder entlang der Z-Ebene aufgenommen, welche die unterschiedlichen Tiefenschichten der Probe abbildeten. Zur zweidimensionalen dieser dreidimensionalen Bilder wurde eine maximale Intensitätsprojektion (MIP) verwendet. Dabei wurde für jedes Pixel der höchste Intensitätswert aus allen Ebenen herangezogen und in einem zweidimensionalen Bild zusammengeführt. Diese Methode hob besonders starke Signale hervor und machte relevante Strukturen besser sichtbar. Gleichzeitig ging jedoch die räumliche Information entlang der Z-Achse verloren. In komplexen Strukturen kann dies dazu führen, dass sich Signale überlagern, verdeckt werden oder nicht eindeutig zugeordnet werden können. Nach der Aufnahme wurden die Bilder zusätzlich mithilfe von Instant Computational Clearing (ICC) verarbeitet. Für die MIP und das ICC wurde die Software LAS X 3.8.1 von Leica verwendet. Weitere Details der Durchführung zeigen (Ahmad et al. 2025).

**Tab. 2: Gennamen und entsprechende Barcodesequenzen der acht markierten viralen Ribonukleoproteinkomplexe des Influenzavirus, die in einer In-situ Sequenzierung untersucht wurden.**

<b>Name</b>	<b>Code</b>
<i>PB2</i>	AGC
<i>PB1</i>	CGC
<i>PA</i>	GGC
<i>HA</i>	TAC
<i>NP</i>	TCC
<i>NA</i>	TTC
<i>M</i>	TGA
<i>NS</i>	TGT



**Abb. 3: Anordnung und Überlappung der FOVs.** Die Abbildung zeigt die Anordnung der FOVs und deren Überlappungen in den Randbereichen exemplarisch für das Replikat null, null Stunden nach Infektion. Die Farbskala zeigt den Grad der Überlappung.

### 3.3 Manuelle Annotation

Eine manuelle Annotation der Bilddaten wurde durchgeführt, um eine Referenzgrundlage für die spätere Auswertung zu erstellen. Insgesamt wurden jeweils zwei zufällig gewählte FOVs pro Infektionszeitpunkt der ersten drei Replikate annotiert. Die Replikate eins und drei wurden jeweils nur null bis sechs Stunden nach Infektion annotiert. Dafür wurden die Bilder eines FOV in Napari geladen (Sofroniew et al. 2025). Napari ist ein interaktiver, auf Python basierender Open-Source-Viewer für multidimensionale Bilddaten. Ein 150x150 Pixel großer Annotationsbereich im Zentrum des Bildes wurde festgelegt. Mithilfe eines selbsterstellten Plug-ins wurden die fluoreszierenden Spots markiert. Durch die Betrachtung des Signals in den unterschiedlichen Kanälen und Sequenzierunden wurde der zugrundeliegende Barcode und das markierte vRNP-Segment bestimmt. Für jeden Spot wurden die detektierten Basen jeder Sequenzierunde durch Kommas getrennt notiert. Wurde in einer Runde kein Signal detektiert, wurde eine null notiert. Sobald das Signal uneindeutig war, wurde eine eins notiert. Bereiche, die so viel Signal enthielten, dass eine Unterscheidung einzelner Spots nicht mehr

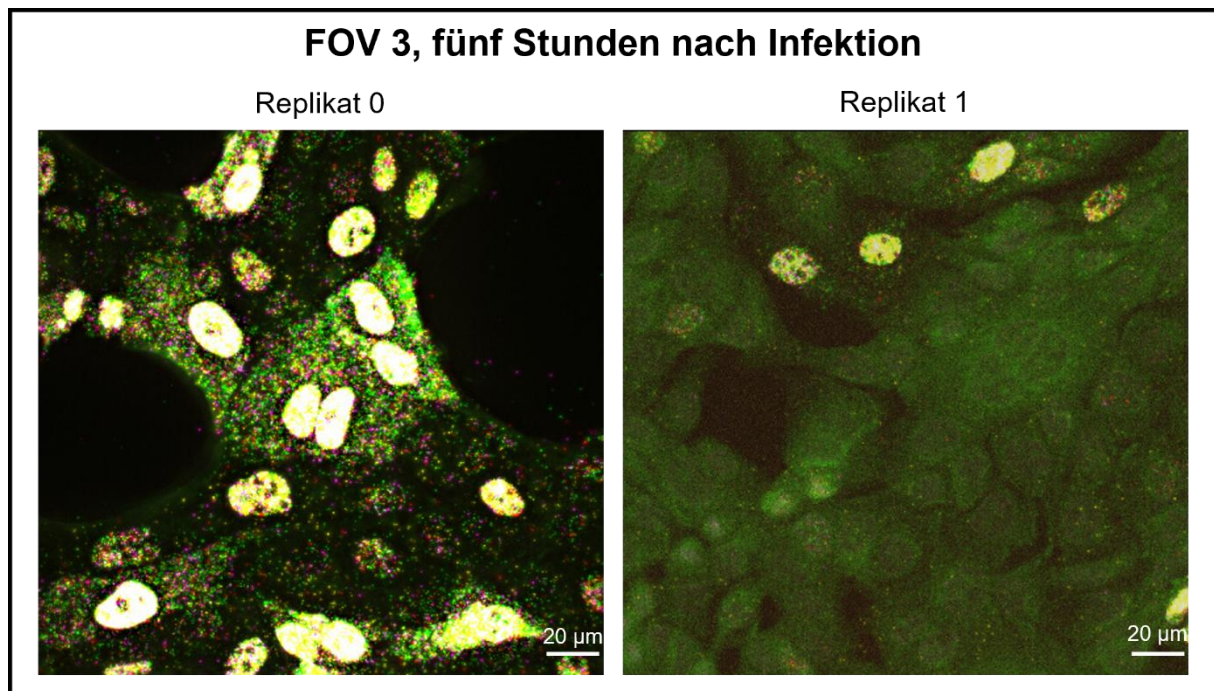
möglich war, wurden ignoriert. Die Koordinaten und die annotierten Basen wurden in einer CSV-Datei exportiert. Das erstellte Plug-in ist verfügbar unter: <https://github.com/janniswi00/np-plugin-manual-spot-annotator>.

Zusätzlich wurde für jeweils zwei FOVs pro Infektionszeitpunkt und Replikat ein Schwellenwert in der maximalen Intensitätsprojektion aller Sequenzierrunden und Kanäle bestimmt. Der Schwellenwert diente der Trennung von Regionen hoher und geringer Signaldichte. Die Schwellenwerte wurden als Perzentilwerte der Intensitätsverteilung definiert und dienten dem Training und der Validierung eines Regressionsmodells, welches den Schwellenwert für die restlichen FOVs festlegte.

### 3.4 Analysepipeline

Die Analysepipeline wurde zum größten Teil mit einer modifizierten Version der Python-Bibliothek Starfish 0.3.2 (Axelrod et al. 2018) implementiert. Diese ist verfügbar unter: [https://github.com/janniswi00/starfish\\_jannis](https://github.com/janniswi00/starfish_jannis). Für die Analyse wurde zunächst mithilfe von *starfish.Experiment.json* das Starfish-Experiment eingelesen, dessen Metadaten in einer json-Datei gespeichert waren. In dem geladenen Experiment waren alle Bilder der FOVs enthalten. Pro FOV lag ein Bildstapel mit den primären Bildern der vier Kanäle und drei Sequenzierrunden vor und ein anderer mit den Bildern des DAPI-Kanals. In **Abb. 4** sind exemplarischere, unbearbeitete Bildausschnitte von den überlagerten Farbkanälen von FOV 3, fünf Stunden nach Infektion dargestellt. Es sind jeweils die Bilder der ersten Sequenzierrunde dargestellt. Links zeigt das Replikat null, rechts das Replikat eins. Wichtig ist zu beachten, dass alle Aufnahmen des Replikats null mit einer anderen Version der internen Software des Mikroskops aufgenommen und im ICC und der MIP verarbeitet wurden. Alle Analyseschritte werden im Folgenden erklärt.





**Abb. 4: Überlagerte Rohbilder der Farbkanäle der ersten Sequenzierrunde von FOV 3, fünf Stunden nach Infektion.** Links zeigt einen Ausschnitt des Replikats null, rechts einen Ausschnitt von Replikant eins. Der Maßstabsbalken zeigt eine Länge von 20 µm.

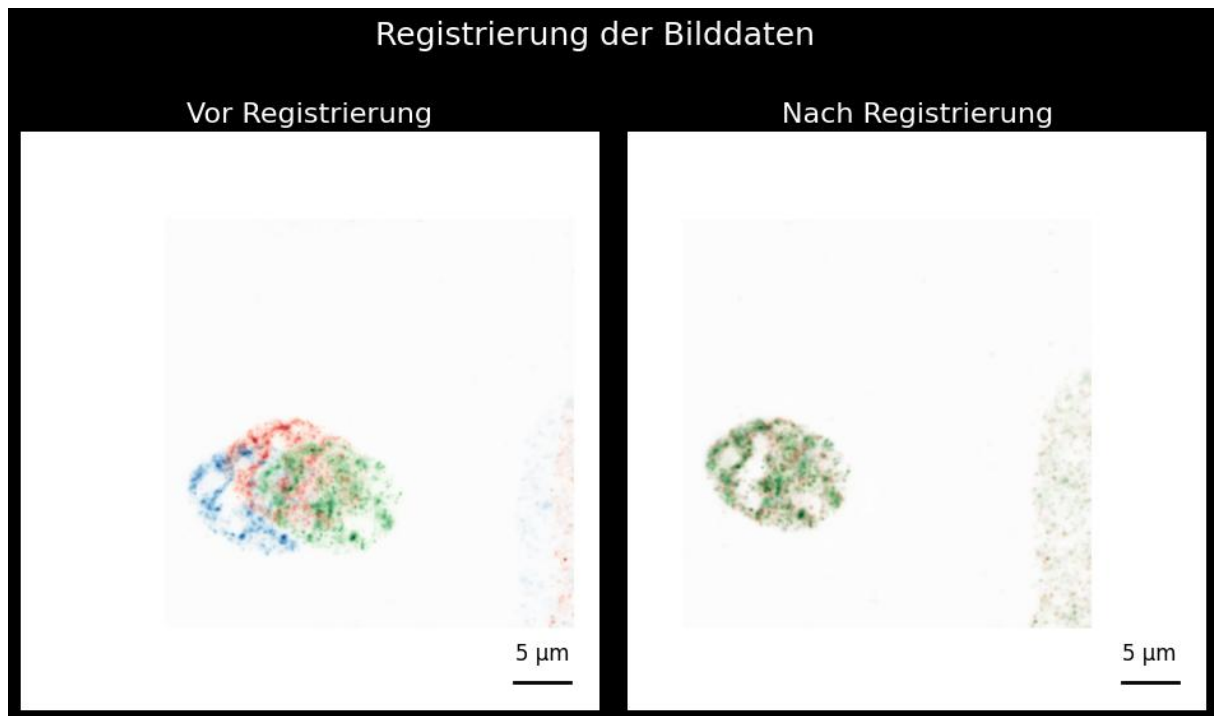
### 3.4.1 Registrierung

Die Registrierung ist ein wichtiger Schritt in der Korrektur von Bildern, vor allem für Experimente, in denen Strukturen über mehrere Kanäle und Sequenzierunden verfolgt werden sollen. Für die Detektion eines Spots musste in drei Sequenzierunden ein Fluoreszenzsignal an der gleichen Position detektiert werden. Daher war es wichtig, dass die Positionen zwischen Sequenzierunden übereinstimmten. Verschiebungen zwischen den Runden konnten beispielsweise durch chromatische Aberrationen, durch die Handhabung des Gewebes während des Experiments oder durch Verschiebungen des Objektträgers zwischen den Runden auftreten.

Zunächst wurde der Bildstapel der primären Bilder reduziert, indem eine Maximalprojektion der Fluoreszenzintensitäten über die Kanalachse erzeugt wurde. Als Resultat ergab sich ein zweidimensionales Bild pro Sequenzierunde, das sämtliche RNA-Signale zusammenfasste. Da dieses Bild alle Signale einer Sequenzierunde enthielt, eignete es sich gut für die Registrierung. Auf die Maximalprojektionen wurde die Funktion *starfish.image.LearnTransform* angewandt, welche die Verschiebung zwischen den Runden berechnete. Als Referenz diente dabei die Projektion der ersten Sequenzierunde. Das Ergebnis war eine Liste der Transformationsparameter, die für jede Sequenzierunde den optimalen x- und y-Versatz, sowie die Rotation und Skalierung zur Referenzrunde enthielten. Mit



`starfish.image.ApplyTransform` konnten die berechneten Verschiebungen auf den ursprünglichen, mehrkanaligen Bildstapel angewandt werden, sodass ein registrierter Bildstapel entstand, in dem alle Sequenzierrunden pixelgenau übereinanderlagen. Die folgende Abbildung (**Abb. 5**) zeigt beispielhaft die Bilder des Replikats null im FOV null, fünf Stunden nach Infektion, links vor der Registrierung und rechts nach der Registrierung.

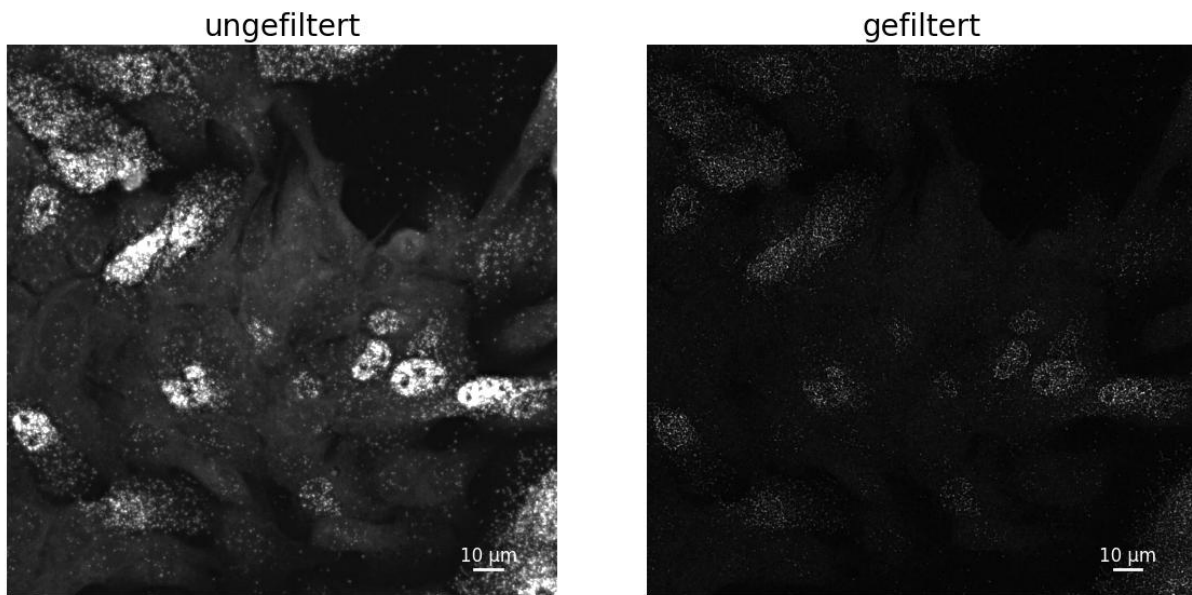


**Abb. 5: Registrierung der Bilddaten über die Sequenzierrunden.** Zur Korrektur von Verschiebungen zwischen den Runden wurde eine translationale Registrierung durchgeführt. Die maximale Intensitätsprojektion über alle Kanäle diente als Grundlage der Registrierung. Die Translation wurde mittels `starfish.image.LearnTransform` durchgeführt und mittels `starfish.util.plot.inspect_registration` visualisiert (Axelrod et al. 2018). Links sind die überlagerten Maximalprojektionen der drei Sequenzierrunden (Runde 1: blau; Runde 2: rot; Runde 3: grün) vor der Registrierung dargestellt. Der Maßstabsbalken zeigt eine Länge von fünf Mikrometern. Rechts zeigt die drei Maximalprojektionen nach der Registrierung. Die Bilder stammen aus Replikat null, FOV null, fünf Stunden nach Infektion.

### 3.4.2 Hintergrundfilterung

Für die Hintergrundfilterung wurde ein White Top-Hat Filter verwendet (Bai et al. 2012). Der Filter ist besonders gut geeignet, um kleine, helle Strukturen hervorzuheben und großflächige, unregelmäßige Hintergrundsignale herauszufiltern. Der Filter wurde mittels `starfish.image.Filter.WhiteTophat` auf die registrierten Bilder angewandt. Die **Abb. 6** zeigt beispielhaft das Bild des Replikats zwei, im FOV vier, der ersten Sequenzierrunde, im Thymin-Kanal, sieben Stunden nach Infektion, links vor der Filterung und rechts nach der Filterung.

## Hintergrundfilterung



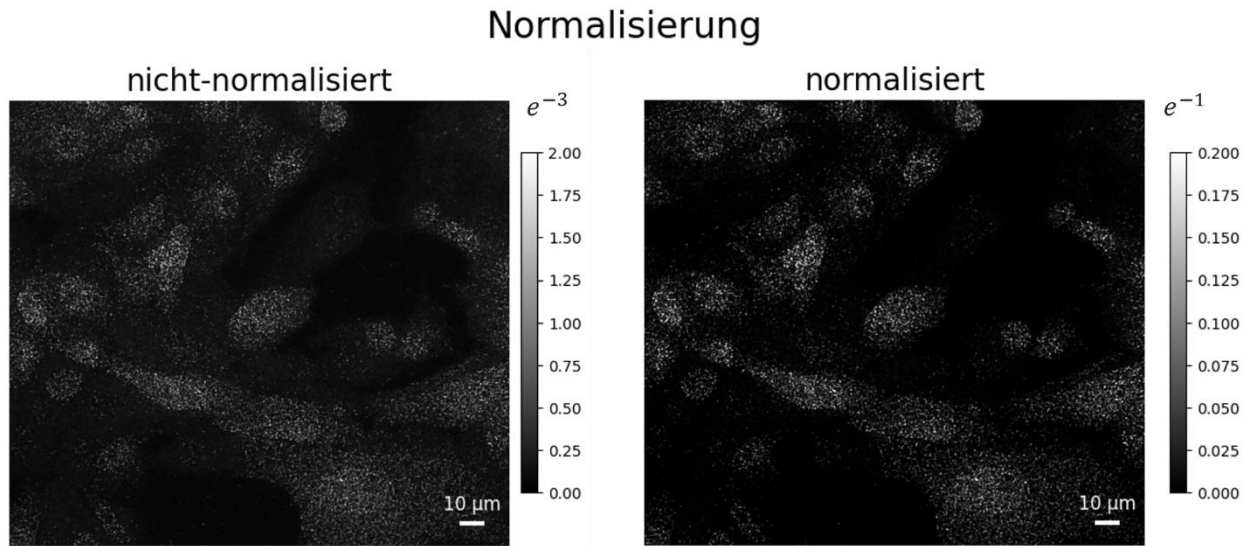
**Abb. 6: Hintergrundfilterung mittels White Top-Hat Filter.** Für die Entfernung von großflächigem Hintergrund wurde ein White Top-Hat Filter angewandt (Bai et al. 2012). Dieser wurde mit `starfish.image.Filter.WhiteTophat` implementiert (Axelrod et al. 2018). Der Maßstabsbalken zeigt eine Länge von zehn Mikrometern. Die Abbildung zeigt das Replikat zwei, im FOV vier, sieben Stunden nach Infektion. Links ist das ungefilterte Bild der ersten Sequenzierrunde im Guanin-Kanal dargestellt, rechts das Ergebnis nach der Hintergrundfilterung.

### 3.4.3 Normalisierung

Nach der Registrierung und Hintergrundfilterung wurden die Fluoreszenzintensitäten normalisiert, um Hintergrundrauschen effektiv zu entfernen und lediglich signalstarke Bildbereiche zu erhalten. Durch diese Normalisierung wurden zudem die Intensitätsskalen über alle Kanäle hinweg vereinheitlicht, was eine konsistente und vergleichbare Signalverarbeitung ermöglichte. Dieser Schritt war wichtig, um in der anschließenden Spot-Detektion zuverlässige Ergebnisse zu erzielen.

Die Normalisierung wurde mit `starfish.image.Filter.ClipPercentileToZero` separat für jeden Kanal durchgeführt. Die Parameter wurden so gesetzt, dass nur die hellsten 20 % der Pixelwerte erhalten blieben, die Intensitäten aller anderen Pixel wurden auf null gesetzt. Zusätzlich wurde mit der Methode `starfish.types.Levels.SCALE_BY_CHUNK` eine automatische Helligkeitsskalierung vorgenommen. Dabei wurden die Intensitätswerte innerhalb jedes Kanals so skaliert, dass der niedrigste Wert auf null und der höchste Wert auf den maximalen Wert des Dynamikbereichs gesetzt wurde. Das glich Unterschiede zwischen den Kanälen aus und auch schwächere Signale konnten besser erkannt werden. Die folgende

Abbildung (Abb. 7) zeigt beispielhaft das Bild des Replikats eins im FOV vier, acht Stunden nach Infektion, in der ersten Sequenzierrunde im Guanin-Kanal, links vor der Normalisierung und rechts nach der Normalisierung. Zur besseren Visualisierung wurden unterschiedliche Farbskalen verwendet.



**Abb. 7: Normalisierung mittels „Percentile Clipping“.** Das Bild wurde mittels *starfish.image.Filter.ClipPercentileToZero* normalisiert (Axelrod et al. 2018). Dabei wurden alle Pixelwerte unterhalb des 80 % Perzentils auf null gesetzt, während die hellsten 20 % der Pixelwerte kanalweise mit einer Min-Max-Normalisierung normalisiert wurden. Der Maßstabsbalken zeigt eine Länge von zehn Mikrometern. Die Abbildung zeigt das Replikat eins, im FOV vier, acht Stunden nach Infektion. Links ist das nicht-normalisierte Bild der ersten Sequenzierrunde im Guanin-Kanal dargestellt, rechts das Ergebnis nach der Normalisierung. Für eine bessere Visualisierung wurden unterschiedliche Farbskalen verwendet.

### 3.4.4 Maskierung signalintensiver Regionen

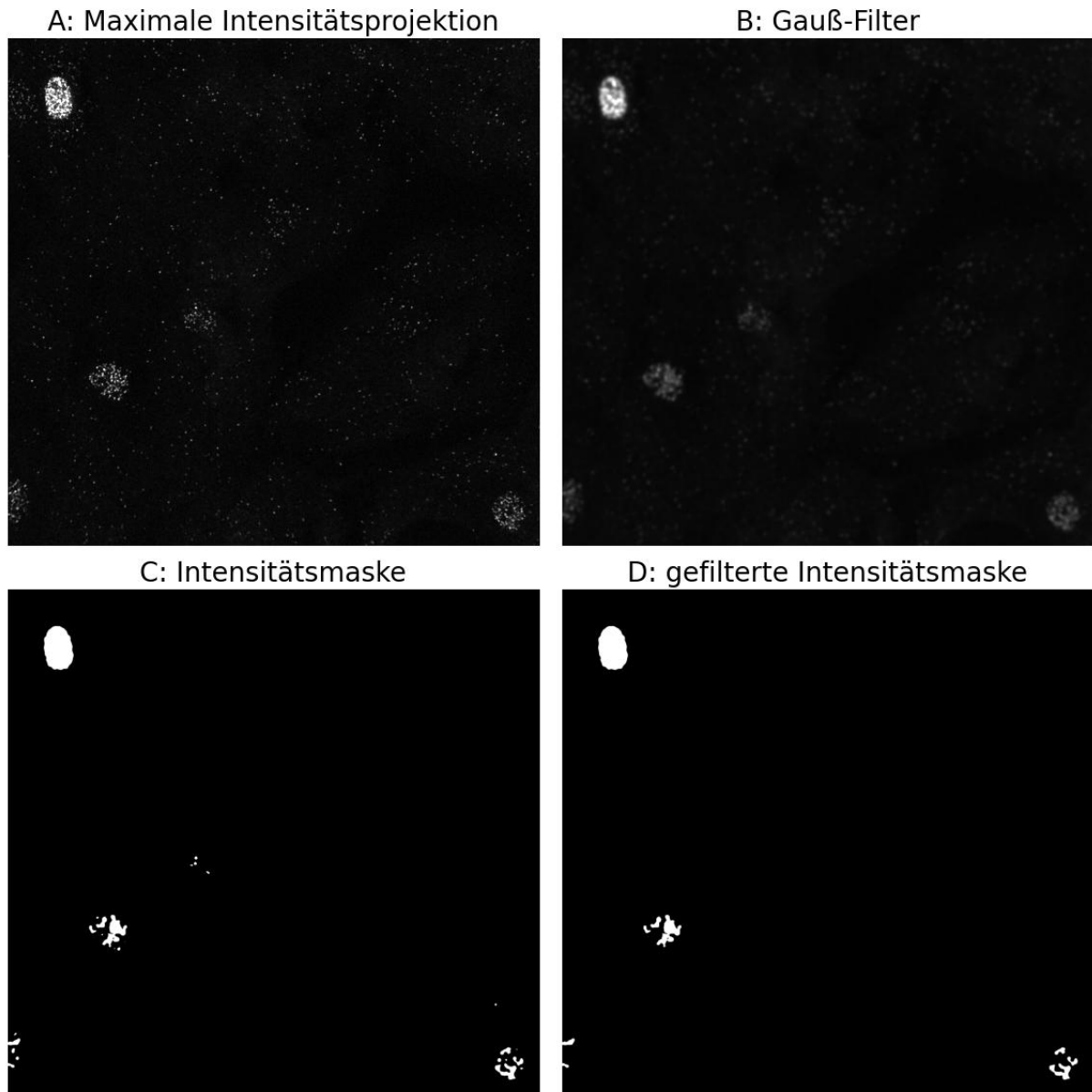
Zur Erstellung einer Maske, die Regionen mit geringer Fluoreszenzintensität von solchen mit hoher Signalstärke trennt, wurde eine Dichtemessung der Fluoreszenzintensitäten durchgeführt. Im Folgenden werden Regionen hoher Signalintensitäten als „dicht“ und Regionen geringer Intensitäten als „undicht“ bezeichnet. Die Maskierung war essenziell, da im Folgenden die Qualität von zwei Dekodierungsmethoden sowohl in dichten als auch in undichten Regionen bewertet werden sollte.

Zunächst wurde das normalisierte Bild entlang der Sequenzierrunden und Kanäle aufsummiert, sodass ein zweidimensionales Gesamtintensitätsbild entstand (Abb. 8 A). Auf dieses Bild wurde ein Gauß-Filter (D’Haeyer 1989) angewandt, welcher die Intensität jedes Pixels durch einen gewichteten Mittelwert seiner Umgebung ersetzt. Der Gauß-Filter hat somit

eine glättende Wirkung. Der Sigma-Wert, welcher den Grad der Glättung angibt, wurde dabei auf fünf gesetzt (**Abb. 8 B**). Der Filter wurde mit `scipy.ndimage.gaussian_filter` implementiert. Für die Trennung zwischen dichten und weniger dichten Regionen wurde mittels `numpy.percentile` (Harris et al. 2020) ein Schwellenwert auf das Bild angewandt. Die Funktion bestimmte den Intensitätswert unter dessen ein bestimmter Prozentsatz der Pixelwerte lag. Anschließend wurde aus dem Bild eine binäre Maske erzeugt. Alle Pixel, deren Intensität über dem Schwellenwert lagen, wurden auf eins gesetzt und alle Pixel, deren Intensitäten unter dem Schwellenwert lagen, wurden auf null gesetzt. Das Ergebnis war eine Maske mit den höchsten Pixelintensitäten (**Abb. 8 C**). Diese Maske wurde weiter gefiltert, indem zunächst mit `scipy.ndimage.sum` (Virtanen et al. 2020) die Fläche, also die Anzahl an Pixeln, aller maskierten Regionen berechnet wurden. Im Anschluss wurden alle Regionen mit einer Fläche kleiner als 200 Pixel herausgefiltert (**Abb. 8 D**). Durch das Filtern sollte verhindert werden, dass einzelne Spots mit einer hohen Intensität als Region hoher Signaldichte erkannt werden.

Der Prozentsatz für die Bestimmung des Schwellenwerts mittels `numpy.percentile` konnte nicht einheitlich für alle FOVs festgelegt werden, da es große Unterschiede in den Intensitäten und deren Verteilungen zwischen den einzelnen FOVs innerhalb der unterschiedlichen Replikate und Infektionszeitpunkte gab. Daher wurde der Prozentsatz für insgesamt 90 FOVs von unterschiedlichen Replikaten und Infektionszeitpunkten manuell festgelegt. Anschließend wurde ein Regressionsmodell trainiert, um die Prozentsätze der restlichen FOVs zu ermitteln. Dieses wurde mit `sklearn.ensemble.HistGradientBoostingRegressor` (Pedregosa et al. 2012) implementiert. Als Eingabe für das Modell dienten eine Reihe statistischer Kennzahlen, die die Intensitätsverteilungen des Bildes charakterisierten. Als grobes Abbild der Intensitätsverteilung wurden 20 gleich breite Histogrammbins über den gesamten Intensitätsbereich angelegt, welche die Pixelhäufigkeiten enthielten. Weiterhin wurde die minimale, maximale und durchschnittliche Intensität, sowie die Standardabweichung berechnet (`numpy`). Als Maß für die Verteilung dienten Schiefekoeffizient und Wölbung (`scipy.stats.skew, -kurtosis`). Ergänzend dazu wurden das erste Quartil, der Median, das dritte Perzentil und das 90 % Perzentil berechnet. Zuletzt wurden noch die Shannon-Entropie (`scipy.stats.entropy`) und das Signal-to-Background Ratio gebildet, welches als Quotient aus dem Mittelwert des obersten Zehnprozentperzentils und dem Mittelwert des untersten Zehnprozentperzentils definiert ist. Für die Infektionszeitpunkte von null bis drei Stunden wurde das Perzentil pauschal auf 100 % gesetzt, sodass keine dichten Regionen maskiert wurden.

## Intensitätsdichte



**Abb. 8: Maskierung von Regionen hoher Signaldichte.** (A) Maximale Intensitätsprojektion aller Sequenzierrunden und Kanäle. (B) Gezeigt ist das Intensitätsbild (A) durch einen Gauß-Filter (D'Haeyer 1989) geglättet ( $\sigma=5$ ). Der Filter wurde mittels `scipy.ndimage.gaussian_filter` implementiert (Virtanen et al. 2020). C zeigt die Maske mit den 99.3 % der hellsten Intensitäten. Das Perzentil wurde durch das Regressionsmodell bestimmt (Details siehe **Fehler! Verweisquelle konnte nicht gefunden werden.**). In D ist die gefilterte Intensitätsmaske dargestellt, in der zusammenhängende Flächen kleiner als 200 Pixel herausgefiltert wurden, um maskierte Einzelspots auszuschließen. Alle Bilder zeigen das Replikat eins, im FOV fünf, fünf Stunden nach Infektion.

### 3.4.5 Spot-basierte Dekodierung

In diesem Teil der Analysepipeline wurde zunächst eine Spoterkennung durchgeführt. Dafür wurde das normalisierte Bild entlang der Sequenzierrunden und Kanäle aufsummiert und auf den maximalen Intensitätswert reduziert. Dadurch entstand ein Referenzbild, das alle potenziellen Spots mit hoher Intensität enthielt. Anschließend wurde mit *starfish.spots.FindSpots.BlobDetector* eine punktförmige Objekterkennung gestartet. Der sogenannte Blobdetektor suchte nach hellen, rundlichen Strukturen, die typischerweise die RNA-Spots darstellten. Die Parameter *min\_sigma* und *max\_sigma* bestimmten dabei den Größenbereich der erkannten Spots. Sie wurden auf eins bzw. fünf gesetzt. Der Parameter *num\_sigma* legte die Anzahl der Zwischenschritte fest und wurde auf 20 gesetzt. Mit dem *threshold* wurde der Schwellenwert festgelegt, ab welcher Intensität ein Signal als Spot galt.

Im nächsten Schritt erfolgte die Dekodierung der Spots, also die Zuordnung einer oder mehrerer viraler Segmente zu jedem erkannten Spot. In einem Codebook wurden für jedes Segment die entsprechende Abfolge von Fluoreszenzsignalen festgehalten. Dieses wurde mittels *starfish.Codebook.open\_json* geladen. Die Zuordnung der Spots zu den Einträgen innerhalb des Codebooks erfolgte mittels *starfish.spots.DecodeSpots.MultiBarcodeDecoder*. Dabei wurde für jeden Spot geprüft, wie gut seine Intensitätsverteilung über die Sequenzierrunden und Kanäle mit einem Eintrag im Codebook übereinstimmte. Bei der Funktion handelte es sich um eine modifizierte Version von *DecodeSpots.BarcodeDecoder*, die auch in der Lage war mehrere Codebook Einträge zu einem Spot zuzuordnen. Dadurch konnten auch Komplexe aus mehreren Segmenten detektiert werden. Der Parameter *max\_distance* legte die maximal zulässige Abweichung der Fluoreszenzsignale vom Barcode des Codebooks fest. Der Parameter *min\_intensity* bestimmte die mindestens erforderliche Intensität des Spots. Wurde in einer Runde kein Signal detektiert, wurde dem Spot der Barcode „missing“ zugeordnet. Bei einer uneindeutigen Zuordnung des Barcodes wurde der Spot zusätzlich mit „(invalid)“ annotiert.

### 3.4.6 Pixel-basierte Dekodierung

Bei der Pixel-basierten Dekodierung wurden die normalisierten Bilder direkt analysiert. Für jedes Pixel im Bild wurde ein Intensitätsvektor über die Runden und Kanäle erstellt und mit allen Barcodes im Codebook verglichen. Lag die Distanz unter einem bestimmten Schwellenwert, wurde das Pixel als eine bestimmte RNA entschlüsselt. Anschließend wurden die einzelnen dekodierten Pixel zu Spots gruppiert.

In der Analysepipeline wurde eine iterative Pixel-basierte Dekodierung durchgeführt. Das heißt, dass die Dekodierung in mehreren Runden durchgeführt wurde und die Parameter mit jeder Runde verändert wurden. Die Dekodierung wurde mit *starfish.core.spots.DetectPixels.pixel.decoder.MultiBarcodePixelSpotDecoder* durchgeführt, einer veränderten Version von *PixelSpotDecoder*, die in der Lage war mehrere Einträge aus dem Codebook zu einem Pixel zuzuordnen. Insgesamt wurden drei Dekodierungsrunden durchgeführt. In der ersten Dekodierungsrunde wurde der *distance\_threshold* auf 0,25 gesetzt. Dieser Parameter setzte den Schwellenwert für die Distanz fest, den der Intensitätsvektor des Pixels zum Eintrag im Codebook maximal haben durfte. Die Distanz wurde in allen drei Runden als Cosinus-Abstand zwischen den Winkeln der Vektoren definiert. Die Größe der Vektoren wurde als euklidische Norm berechnet. Mit dem Parameter *magnitude\_threshold* wurde ein Schwellenwert für die Intensität festgelegt, die ein Pixel haben musste, um als Signal gewertet zu werden. In der ersten Runde wurde dieser auf 0,30 gesetzt. Mit jeder Dekodierungsrunde wurde der Distanzschwellenwert um 0,05 erhöht und der Intensitätsschwellenwert um 0,075 verringert. Durch die Lockerung der Schwellenwerte mit jeder Iteration, wurden mit jeder Iteration mehr Pixel dekodiert. Nach jeder Dekodierung wurde eine Maske aktualisiert, die bereits dekodierte Pixel enthielt und nach jeder Runde auf das Bild angewandt wurde, sodass keine Pixel mehrfach dekodiert wurden. Der Vorteil der iterativen Dekodierung ist der gute Kompromiss zwischen der Größe detektierten Spots und der Anzahl erfasster Spots. Während eine globale Einstellung einerseits oft zu wenig und übermäßig großen Spots, andererseits zu vielen und kleinen Spots führte, ermöglichte die iterative Methode eine Detektion verschiedenster Spots. In frühen Iterationen wurden intensive Pixel dekodiert, während in späteren Runden auch schwächere Pixel dekodiert wurden. Insgesamt wurden drei Iterationen durchgeführt.

Nach der Dekodierung der Pixel erfolgte die Gruppierung der Pixel zu Spots mittels *sklearn.cluster.DBSCAN*. Die Parameter *eps* und *min\_samples* bestimmten die maximale Distanz zwischen zwei Pixeln und die Mindestanzahl an Pixeln pro Spot. Sie wurden auf eins bzw. drei gesetzt. Für jeden Spot wurde das Zentrum und der Radius berechnet. Als Nächstes wurde ermittelt, wie viele Pixel pro Spot mit welchem Barcode vorlagen, indem für jeden Barcode der relative Anteil an der Gesamtpixelzahl des Spots berechnet wurde. Barcodes, die weniger als 10 % der Spotpixel ausmachten, wurden verworfen. Damit wurde sichergestellt, dass nicht einzelne falsch dekodierte Pixel die Dekodierung des gesamten Spots verfälschten. Zuletzt wurden Spots mit einem Radius kleiner als einem Pixel und größer als zwölf Pixel verworfen.

### 3.4.7 Zellkernsegmentierung

Der Bildstapel mit den Bildern des DAPI-Kanals wurde entlang der Sequenzierrunden- und Kanalachse reduziert, sodass eine zweidimensionale Maximalprojektion entstand. Das Bild wurde mit *starfish.image.Filter.GaussianLowPass* geglättet, um Rauschen und feine Artefakte zu reduzieren. Mithilfe von *skimage.filters.threshold\_otsu* wurde ein Schwellenwert festgelegt, der helle Bereiche der Zellkerne vom Hintergrund trennte. Anschließend wurde mit *starfish.morphology.Binarize.ThresholdBinarize* eine binäre Nukleusmaske erstellt. Um zusammenhängende Nukleusbereiche voneinander zu trennen, wurden diese mittels *starfish.morphology.Filter.MinDistanceLabel* zentriert und als Marker beschriftet. Die Nukleusbereiche mussten einen Mindestabstand von 56 Pixeln zueinander haben. Zusätzlich wurde mit der Funktion *starfish.morphology.Filter.AreaFilter* nach Markern mit einer Größe zwischen 1.000 und 100.000 Pixeln gefiltert.

In einer anderen Arbeit wurden die gesamten Zellen und Zellkerne detaillierter segmentiert (Joél Schaust 2024).

## 3.5 Berechnung von Komplexwahrscheinlichkeiten

Für die Berechnung der Komplexwahrscheinlichkeiten wurden zunächst die relativen Abundanzen der Einzelsegmente innerhalb der Replikate und Infektionszeitpunkte berechnet, wobei nur die Infektionszeiten vier bis sechs und undichte Regionen betrachtet wurden. Es wurden sowohl die Einzelsegmente als auch die Segmente, die im Komplex vorlagen, gezählt. Anschließend ergaben sich die Komplexwahrscheinlichkeiten durch die Multiplikation der relativen Abundanzen der Einzelsegmente und deren Normalisierung innerhalb der Komplexgrößen. Zuletzt wurde das arithmetische Mittel der Komplexwahrscheinlichkeiten über die jeweils fünf Replikate der Infektionszeitpunkte vier bis sechs gebildet ( $n = 15$ ). Zudem wurden für relative Abundanz und Wahrscheinlichkeit der Standardfehler berechnet. Zuletzt wurde mittels zweiseitigem, unabhängigen T-Test (*scipy.stats.ttest\_ind*) auf signifikante Unterschiede zwischen der beobachteten relativen Abundanz und der berechneten Komplexwahrscheinlichkeit getestet ( $\alpha = 0,05$ ).

## 3.6 Modellierung der Genomassemblierung

Für die Modellierung der Assemblierung der acht vRNP-Segmente wurde ein monomerer Mechanismus angenommen. Auf Grundlage der relativen Häufigkeiten der einzelnen Segmente wurden die Wahrscheinlichkeiten aller möglichen Assemblierungswege berechnet.



Dafür wurden zunächst die relativen Abundanzen aller einzigartigen Komplexe berechnet, wobei alle Komplexe von vier bis sechs Stunden nach Infektion in undichten Regionen betrachtet wurden. Danach wurden alle möglichen 40.320 Permutationen der acht Segmente mittels *itertools.permutations* (van Rossum 2020a) berechnet. Für jede Permutation wurde eine schrittweise Assemblierung simuliert, bei der ausgehend vom ersten bis achten Segment nach und nach Komplexe gebildet wurden. Für jeden dieser Zwischenkomplexe wurde die logarithmierte relative Häufigkeit berechnet. Die Summe aller Logarithmen diente als Maß für die Wahrscheinlichkeit der jeweiligen Permutation als möglicher Assemblierungspfad.

## 4. Ergebnisse

### 4.1 Evaluation der Analysepipeline

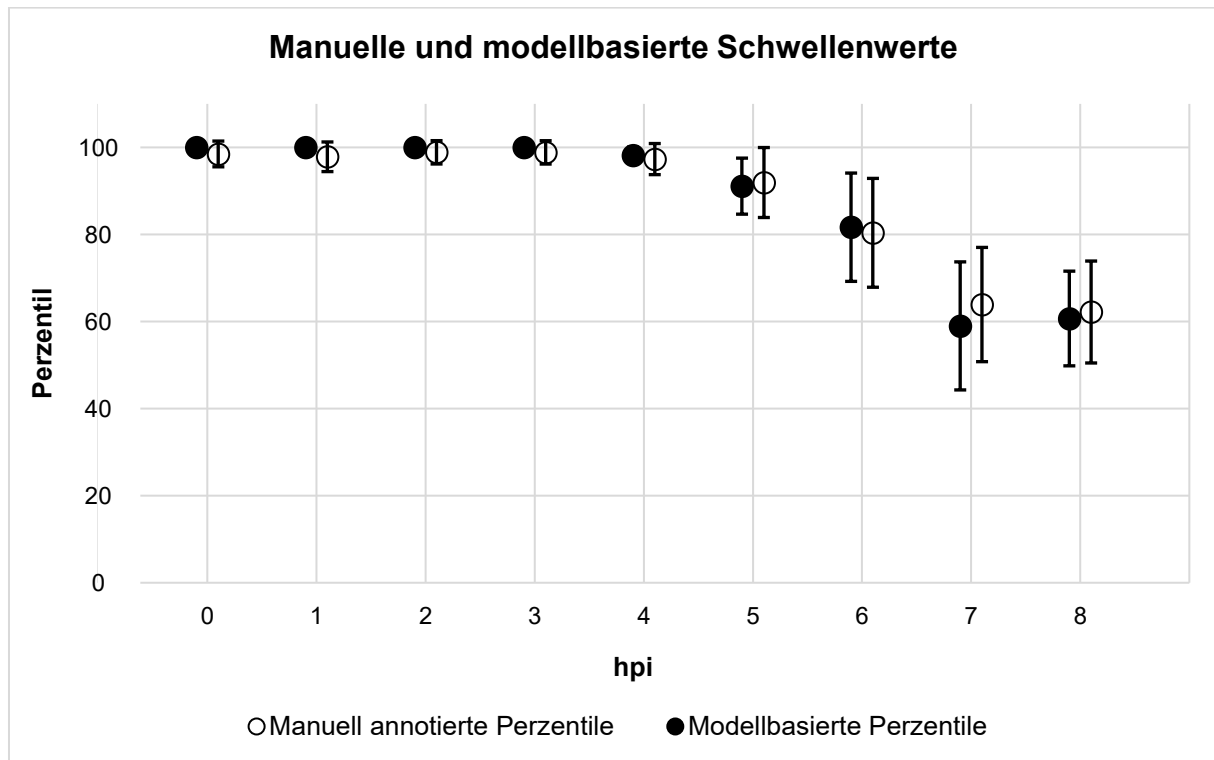
In den folgenden Analysen wurde die gesamte Analysepipeline, sowie Spot-basierte Dekodierung (SBD) und Pixel-basierte Dekodierung (PBD) evaluiert. Die Validierung erfolgte über den Vergleich mit manuell erstellten Referenzen oder durch direkte Vergleiche der Methoden.

#### 4.1.1 Modellbasierte Maskierung

Für die Bilder sollte eine automatisierte Maskierung erstellt werden, die Regionen hoher Signaldichte von Regionen niedriger Signaldichte trennt, um unterschiedliche Dekodierungsstrategien sowohl in dichten als auch undichten Regionen zu validieren. Als Schwellenwert für die Trennung konnte kein einheitliches Perzentil festgelegt werden. Daher wurde das Perzentil für einige FOVs manuell festgelegt, welche dann mit einer Reihe statistischer Kennzahlen als Eingabe für ein Regressionsmodell dienten. Basierend darauf bestimmte das Modell die Perzentile der restlichen FOVs. In **Abb. 9** sind die manuell bestimmten (schwarz) und modellbasierten Perzentile (grau) im Vergleich dargestellt. Die Abbildung zeigt jeweils das arithmetische Mittel der Perzentile aller FOVs einer Infektionszeit und die Standardabweichung. Bei der manuellen Annotation betrug die Anzahl an annotierten FOVs pro Infektionszeitpunkt zehn und bei der modellbasierten Annotation 80 bis 81.

In den ersten vier Infektionszeitpunkten betrug das manuell bestimmte Perzentil 100 %. Mit zunehmender Infektionszeit wurde das Perzentil kleiner, sodass es bei sieben und acht Stunden nach Infektion bei circa 60 % lag. Zugleich wurde die Varianz der Perzentile immer größer, was an der zunehmenden Standardabweichung zu sehen war. Die modellbasierten Perzentile waren im Vergleich zu den manuell annotierten Perzentilen sehr ähnlich. Es zeigten sich nur geringfügige Abweichungen in den Mittelwerten und Standardabweichungen. Im

Durchschnitt betrugen die Abweichungen zwischen manueller und modellbasierter Perzentile 1,7 %.



**Abb. 9: Vergleich von manueller und modellbasierter Annotation der Schwellenwerte zur Trennung dichter und undichter Regionen.** Für die Trennung wurden Perzentile in den Intensitätsverteilungen der maximalen Intensitätsprojektionen bestimmt. Die Bestimmung erfolgte für insgesamt 90 FOVs manuell. Mit den manuell annotierten Perzentilen und einer Reihe statistischer Kennzahlen wurde ein Regressionsmodell trainiert, das die Schwellenwerte für die restlichen FOVs bestimmte (siehe **Fehler! Verweisquelle konnte nicht gefunden werden.**). Dargestellt sind die arithmetischen Mittel der Schwellenwerte aller FOVs pro Infektionszeitpunkt. Die schwarzen Punkte zeigen die manuell gewählten Perzentile ( $n = 10$ ) und die weißen Punkte die vom Modell vorhergesagten Perzentile ( $n = 80-81$ ). Die Fehlerbalken zeigen die Standardabweichung.

## 4.1.2 Spot-basierte Dekodierung

### 4.1.2.1 Vergleich der Spotanzahlen

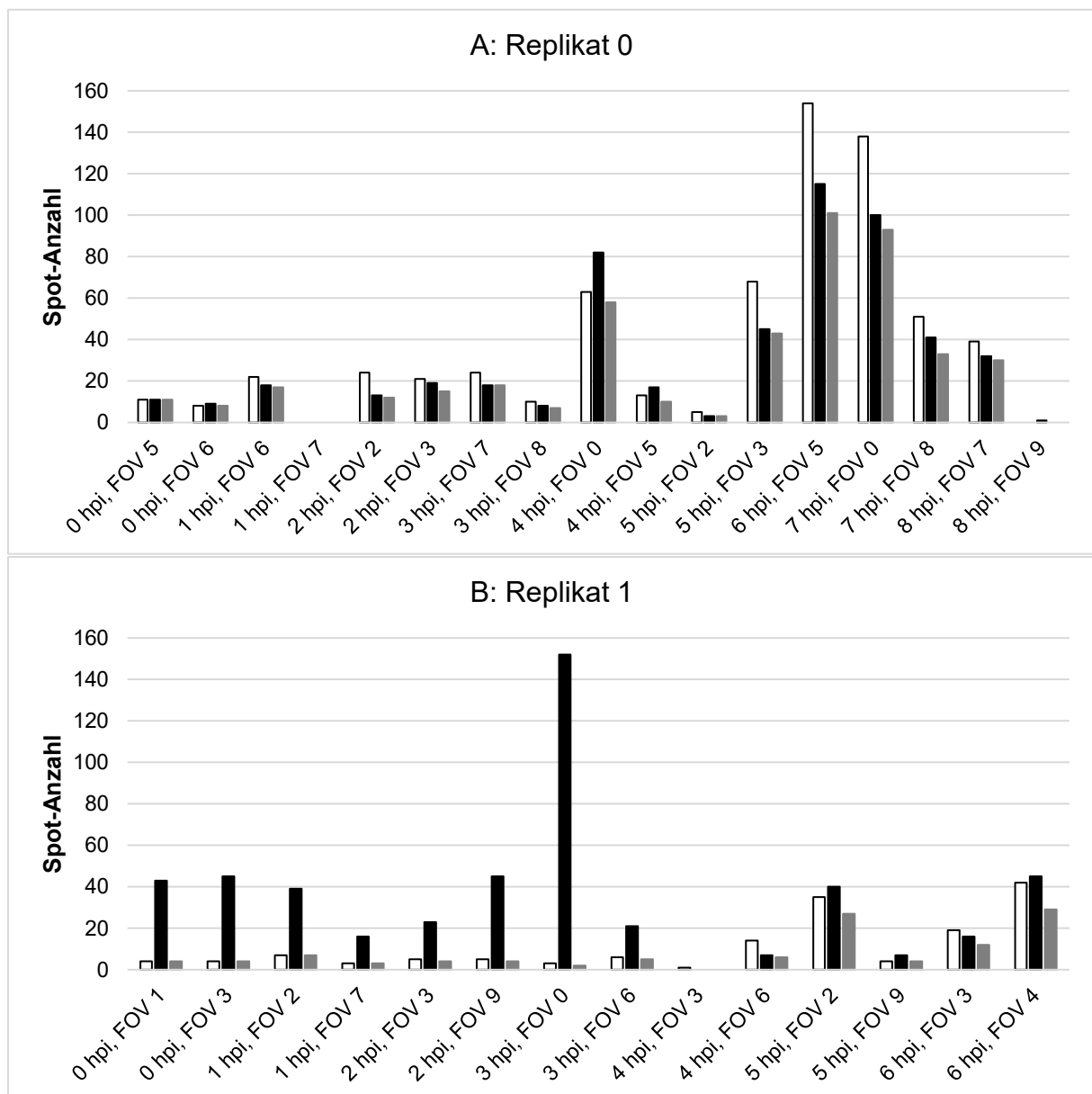
Für die Bewertung der Pipeline wurde die Anzahl an detektierten Spots pro FOV mit der Anzahl an manuell annotierten Spots verglichen. Die Spotanzahlen der manuellen Annotation dienten dabei als Referenz. In **Abb. 10** sind die absoluten Spotanzahlen der manuellen Annotation (weiß) und der Pipelineannotation (schwarz) pro FOV dargestellt. In grau ist die Anzahl der Spots dargestellt, die sowohl manuell als auch von der Pipeline annotiert wurden dargestellt. Die Spots galten als übereinstimmend, wenn die euklidische Distanz weniger als fünf Pixel betrug. Die Abbildung zeigt die Spots der Spot-basierten Dekodierung aus den undichten

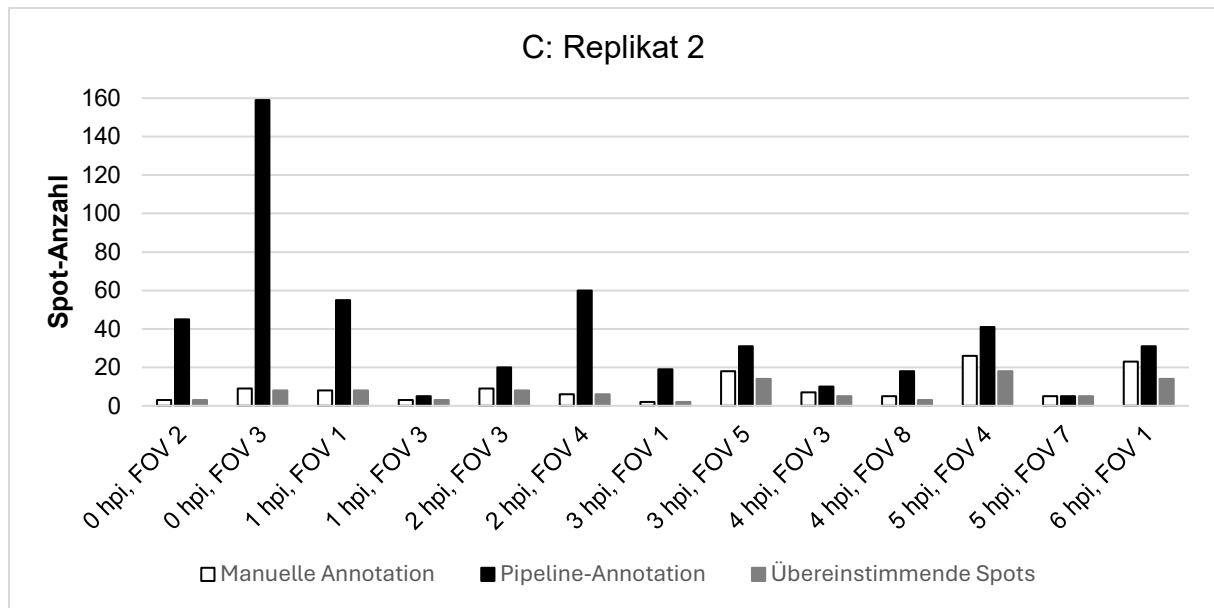
Regionen. Die Spots der undichten Regionen machten einen Anteil von 70 % an allen Spots des Annotationsbereichs aus. Insgesamt wurden im undichten Annotationsbereich 927 Spots manuell annotiert und 1530 durch die Pipeline. Von diesen Spots wurden 667 Spots übereinstimmend durch beide Annotationen detektiert.

Die Ergebnisse des Replikats null (A) zeigen, dass es in den meisten FOVs keine großen Unterschiede in den Spotanzahlen zwischen manueller Annotation und Pipelineannotation gab. Meistens lag die Anzahl der manuell annotierten Spots etwas über der Anzahl der Pipelineannotation. Die Anzahl der übereinstimmenden Spots war meistens etwa so groß wie die niedrigere Anzahl der beiden Annotationen. In „1 hpi, FOV 7“ wurden sowohl durch die manuelle Annotation als auch durch die Pipelineannotation keine Spots detektiert. In den FOVs „5 hpi, FOV 3“, „6 hpi, FOV 5“ und „7 hpi, FOV 0“ war die Spotanzahl der manuellen Annotation 30 % bis 50 % höher als die Anzahl der Pipelineannotation. Im Gegensatz dazu war in „4 hpi, FOV 0“ die Spotanzahl der Pipelineannotation um 30 % höher als die Anzahl der manuellen Annotation.

Bei Replikat eins (B) waren die Spotanzahlen der Pipelineannotation, bis auf „4 hpi, FOV 6“, höher als bei der manuellen Annotation. Diese Unterschiede waren vor allem zu den Infektionszeitpunkten null bis drei extrem. Die Spotanzahlen der Pipelineannotation überstiegen die Anzahl der manuellen Annotation um das 5- bis 50-fache. In „3 hpi, FOV 0“ beispielsweise betrug die Spotanzahl der manuellen Annotation drei und die der Pipelineannotation 152. Die Anzahl der übereinstimmenden Spots war in allen FOVs zu den Infektionszeiten null bis fünf Stunden nach Infektion nahezu identisch mit der Anzahl der manuellen Annotation. Lediglich die beiden FOVs „6 hpi, FOV 3“ und „6 hpi, FOV 4“ zeigten eine deutlich geringere Anzahl an übereinstimmenden Spots.

Die Anzahlen der Pipelineannotation von Replikat zwei (C) zeigten, genau wie Replikat eins, deutlich größere Werte als die Anzahlen der manuellen Annotation. Am größten war der Unterschied in „0 hpi, FOV 3“ mit neun Spots in der manuellen Annotation und 159 Spots in der Pipelineannotation. Außerdem waren auch in diesem Replikat die Anzahlen der übereinstimmenden Spots zu den Infektionszeitpunkten null bis fünf nahezu identisch mit der Anzahl der manuellen Annotation. Zudem war ebenfalls nur die Anzahl der übereinstimmenden Spots zum Infektionszeitpunkt sechs geringer als die Anzahl der manuell annotierten Spots.



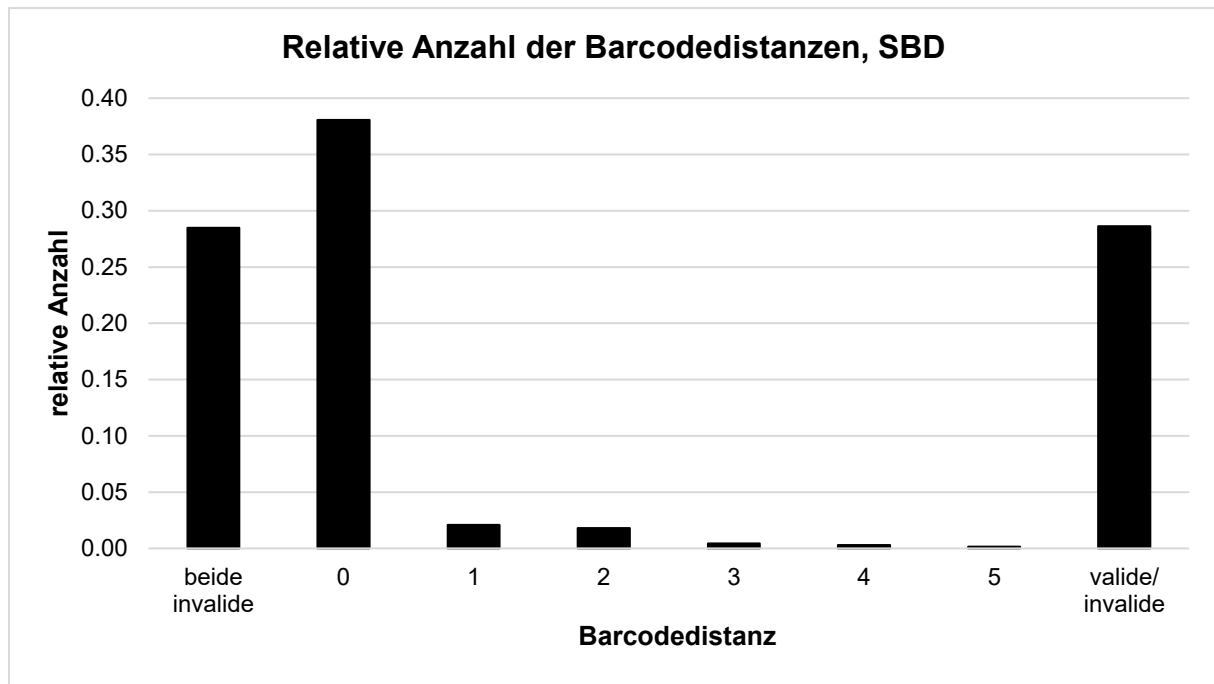


**Abb. 10: Vergleich der Spotanzahlen von manueller Annotation und Pipelineannotation (Spot-basierte Dekodierung) für drei Replikate.** Die Säulen zeigen die absoluten Spotanzahlen pro FOV. Spots, die ausschließlich manuell annotiert wurden, sind weiß dargestellt, Spots aus der Pipelineannotation schwarz. Spots, die in beiden Methoden detektiert wurden, sind grau dargestellt. Die Abbildung zeigt nur Spots der Spot-basierten Dekodierung aus undichten Regionen.

#### 4.1.2.2 Barcodedistanz

Da die Betrachtung der Spotanzahlen keine Aussage über die korrekte Dekodierung der Spots liefert, wurde die Barcodedistanz zwischen den übereinstimmenden Spots aus **Abb. 10** berechnet. Eine Barcodedistanz von zwei beispielsweise bedeutete, dass sich der Spot der Pipelineannotation in zwei Basen von dem Spot der manuellen Annotation unterschied. Wurden beide Spots eines Spotpaares als invalide annotiert, wurde dem Paar die Distanz „beide invalide“ zugewiesen. Ein Spotpaar aus einem validen und einem invaliden Spot wurde als „valide/invalide“ gekennzeichnet.

**Abb. 11** zeigt die relativen Anzahlen der Spotpaare mit einer bestimmten Barcodedistanz. Der Anteil an Spotpaaren, bei denen beide Spots als invalide annotiert wurden, betrug 28 %. Spotpaare mit übereinstimmenden Barcodes, also einer Distanz von null, machten einen Anteil von 38 % aus. Dieser Anteil war somit der höchste. Spotpaare mit einer Distanz zwischen eins und fünf hatten jeweils einen Anteil von zwei Prozent oder weniger. Spotpaare, die aus einem validen Spot und einem invaliden Spot bestanden, machten ebenfalls einen Anteil von 28 % aus. Damit machte der Gesamtanteil von Spots, die manuell und durch die Pipeline gleich annotiert wurden, einen Anteil von 67 % aus.



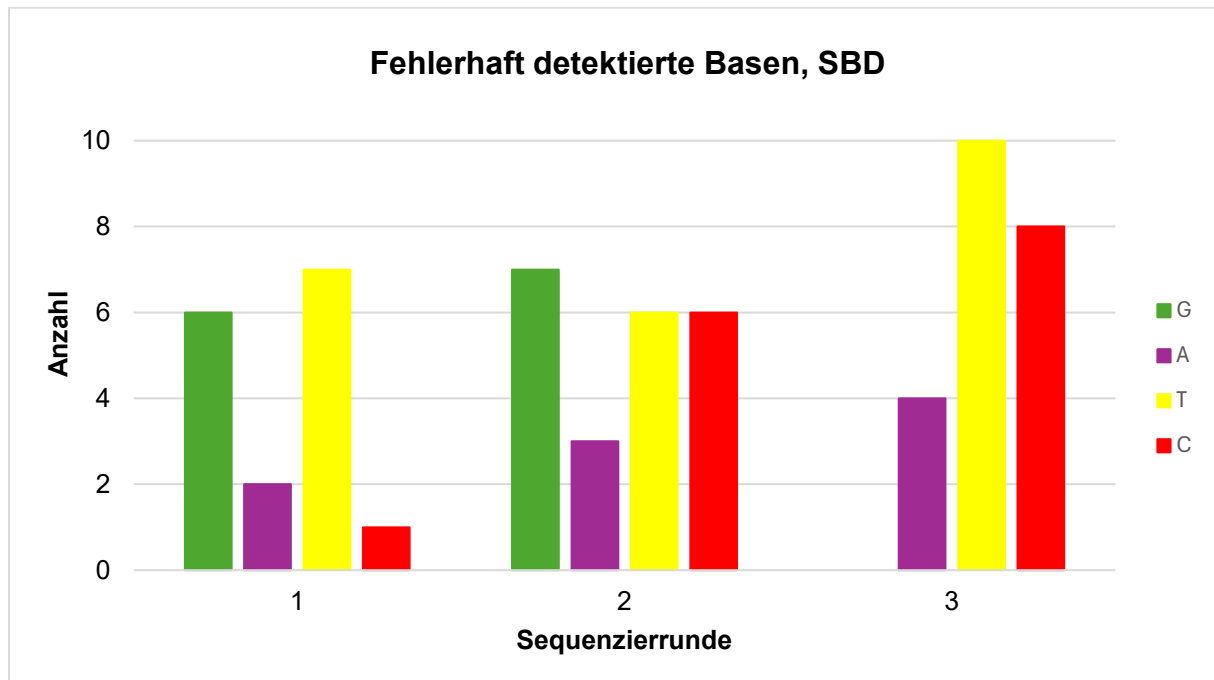
**Abb. 11: Barcodedistanzen zwischen Spotpaaren der manuellen Annotation und Pipelineannotation (Spot-basierte Dekodierung).** Insgesamt wurden jeweils 667 Spots aus manueller Annotation und Pipelineannotation, mit einer euklidischen Distanz kleiner als fünf Pixel, gepaart. Für jedes Spotpaar wurde die Barcodedistanz als Summe der unterschiedlichen Basen berechnet. Die Säulen zeigen die relativen Anzahlen der Spotpaare mit einer bestimmten Barcodedistanz. Es sind die relativen Anzahlen der Distanz null bis fünf, „beide invalide“ und „valide/invalide“ dargestellt. „beide invalide“ stellt Spotpaare dar, die sowohl in der manuellen als auch in der Pipelineannotation invalide waren. „valide/invalide“ stellt Spotpaare dar, bei denen ein Spot als valide und ein Spot als invalide annotiert wurde. Die Spots der Pipelineannotation wurden mittels Spot-basierter Dekodierung in undichten Regionen detektiert.

#### 4.1.2.3 Fehlerhafte Detektion von Einzelbasen

In der folgenden Analyse wurde beurteilt, ob es systematische Fehler in der Pipelineannotation gab. Dafür wurden die Barcodes der 32 Spotpaare mit einer Barcodedistanz von eins bis fünf aus **Abb. 11** verglichen. Für jede Sequenzierrunde wurde die Anzahl von fehlerhaft detektierten Basen in der Pipelineannotation bestimmt. Die manuelle Annotation diente als Referenz.

Die **Abb. 12** zeigt, dass in der ersten Runde die Basen Guanin und Thymin, mit sechs- und siebenmal, am häufigsten falsch detektiert wurden. Die Basen Adenin und Cytosin wurden dagegen nur zwei bzw. einmal falsch detektiert. In der zweiten Runde war die Anzahl von Thymin und Cytosin mit sechs falsch detektierten Basen gleich. Guanin zeigte mit sieben Basen die höchste Anzahl und Adenin mit drei Basen die niedrigste. In der dritten Runde wurde Guanin kein einziges Mal falsch detektiert, während die Anzahlen von Adenin, Thymin und

Cytosin jeweils die höchsten Werte der drei Runden darstellten. Insgesamt waren die Anzahlen zwischen den Basen ähnlich und es konnten keine größeren Ausreißer identifiziert werden.



**Abb. 12: Anzahl an fehlerhaft annotierten Basen durch Spot-basierte Dekodierung in undichten Regionen.** Für die insgesamt 32 gepaarten Spots mit einer Barcodedistanz von eins bis fünf aus Abb. 12 wurden die Anzahlen der fehlerhaft detektierten Basen pro Sequenzierunde bestimmt. Guanin ist in Grün dargestellt, Adenin in Violett, Thymin in Gelb und Cytosin in Rot.

#### 4.1.2.4 Konsistenz in überlappenden FOV-Bereichen

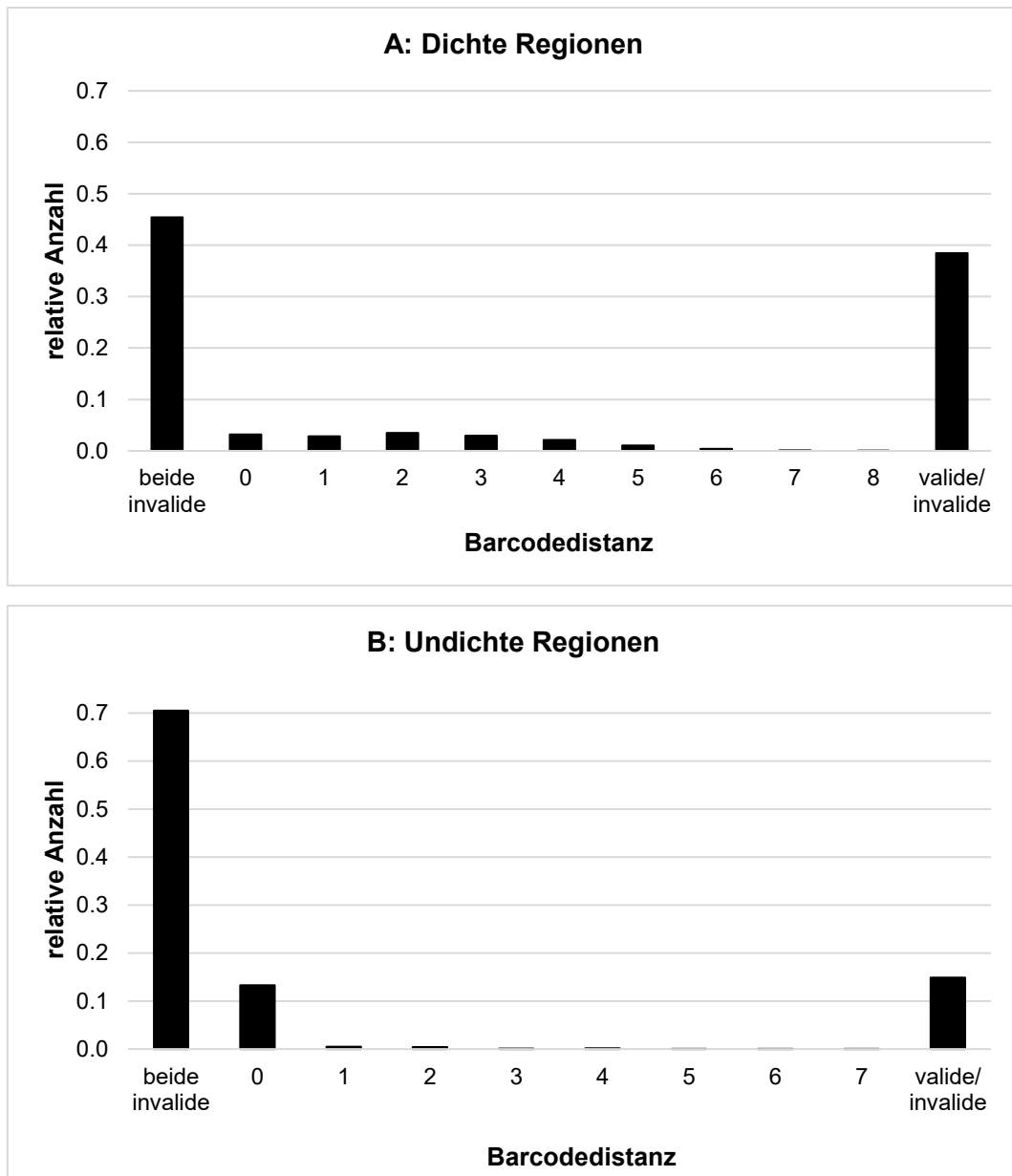
Um die konsistente Annotation der Pipeline anhand einer großen Anzahl von Spots zu prüfen, wurden die Spots aus den überlappenden FOV-Bereichen analysiert. Durch die Überlappung der FOVs in den Randbereichen (**Abb. 3**) wurden einige Regionen mehrfach aufgenommen. Das ermöglichte den Vergleich der Pipelineannotationen von gleichen Spots aus unterschiedlichen Aufnahmen und die Überprüfung der konsistenten Dekodierung. Dafür wurden zunächst die überlappenden Bereiche der FOVs ausgeschnitten und erneut registriert. Dies war nötig, damit kleine Verschiebungen zwischen den FOVs behoben werden konnten und die Spotkoordinaten angeglichen werden konnten. Nach der Registrierung wurden die überlappenden Bereiche auf die gemeinsame Region zugeschnitten. Alle Spots der gemeinsamen Region wurden aus beiden FOVs extrahiert und gepaart, wenn die euklidische Distanz weniger als fünf Pixel betrug. Im Anschluss wurde die Barcodedistanz zwischen den Spots beider FOVs berechnet (siehe 4.1.2.2). In **Abb. 13** sind jeweils die relativen Anzahlen der Spotpaare mit einer bestimmten Barcodedistanz dargestellt. In (A) wurden alle Spots aus dichten Regionen, in (B) alle Spots aus undichten Regionen analysiert.

Insgesamt wurden in den überlappenden Bereichen beider FOVs 281.141 Spots in dichten Regionen detektiert. Davon konnten jeweils 80.868 Spots gepaart werden. Die Barcodedistanzen und die jeweiligen Anteile sind in (A) dargestellt. Den größten Anteil machten Spotpaare aus, bei denen die Spots in beiden FOVs invalide waren. Der Anteil betrug 45 %. Der Anteil von Spotpaaren mit einem validen und einem invaliden Spot war mit 39 % etwas geringer. Spotpaare mit übereinstimmenden Barcodes (Distanz null) machten einen Anteil von drei Prozent aus. Spotpaare mit Distanzen zwischen eins und vier machten jeweils einen Anteil von zwei bis vier Prozent aus. Die Anteile der Distanzen fünf bis acht lagen bei einem Prozent oder weniger.

In den undichten Regionen wurden insgesamt 287.851 Spots detektiert, wovon 51.528 Spots gepaart werden konnten. Die Barcodedistanzen und deren Anteile sind in (B) dargestellt. Mit 70 % war der Anteil an Spotpaaren mit zwei invaliden Spots am höchsten und 25 % höher als in den dichten Regionen (A). Spotpaare mit übereinstimmenden Spots (Distanz null) und Spotpaare mit einem validen und einem invaliden Spot hatten, mit 13 % und 15 %, einen vergleichbaren Anteil. Im Vergleich zu den dichten Regionen war der Anteil der Distanz null zehn Prozent höher. Der Anteil von Spotpaaren mit einem validen und einem invaliden Spot betrug weniger als die Hälfte (15 %). Die Anteile der Distanzen eins bis sieben waren sehr gering. Sie lagen bei einem Prozent oder weniger. Die Anteile der Distanzen eins bis vier in den dichten Regionen waren mit zwei bis vier Prozent deutlich höher. Im Gegensatz zu den dichten Regionen wurden keine Spotpaare mit einer Distanz von acht detektiert.

Insgesamt waren die Anteile von Spotpaaren mit übereinstimmenden Barcodes („beide invalide“ und Distanz null) in den undichten Regionen 35 % höher als in den dichten Regionen. Zudem waren die Anteile von nicht übereinstimmenden Barcodes (Distanz eins bis acht und „valide/invalide“) in den undichten Regionen niedriger.





**Abb. 13: Barcodedistanzen zwischen Spotpaaren aus überlappenden FOV-Bereichen (Spot-basierte Dekodierung).** Überlappende FOV-Bereiche wurden ausgeschnitten, registriert und Spots beider FOVs wurden aus den Bereichen extrahiert. Spots aus unterschiedlichen FOVs aus dem Überlappungsbereich, deren euklidische Distanz weniger als fünf Pixel betrug, wurden gepaart. Berechnung der Barcodedistanz erfolgte wie in 4.1.2.2. Beide Abbildungen zeigen die relativen Anteile der Spotpaare mit einer bestimmten Barcodedistanz. (A) Dargestellt sind insgesamt 80.868 Spotpaare aus dichten Regionen. (B) Dargestellt sind 51.528 Spotpaare aus undichten Regionen.

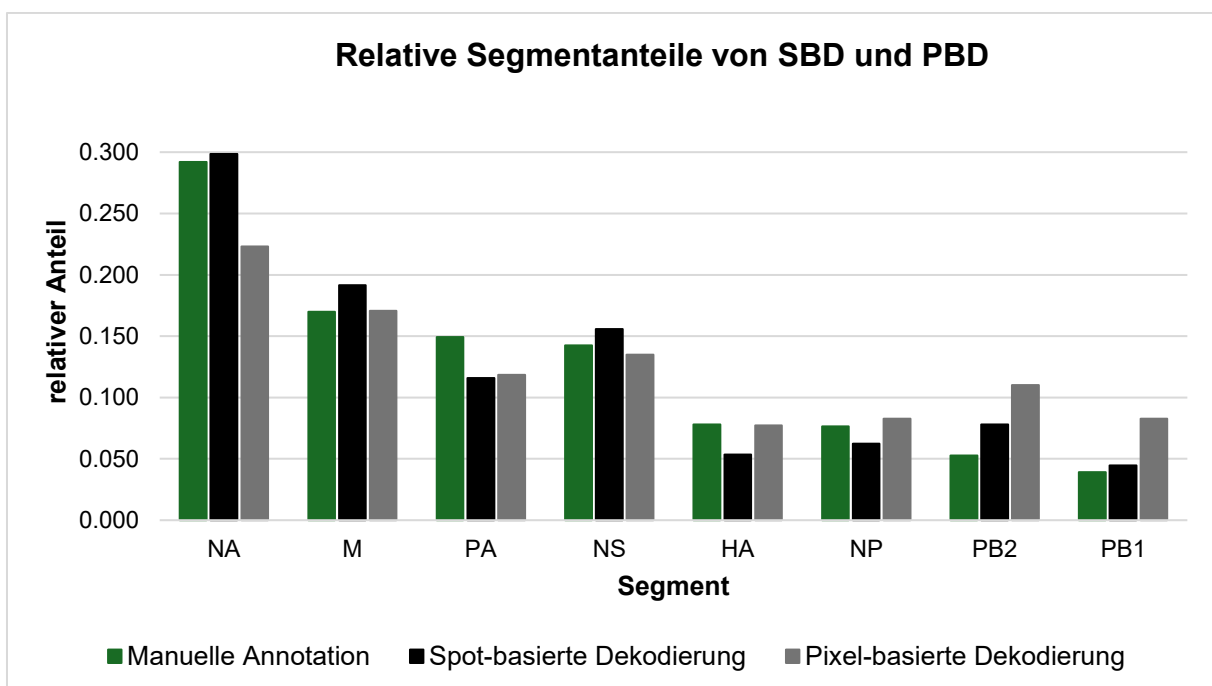
### 4.1.3 Pixel-basierte Dekodierung

#### 4.1.3.1 Vergleich relativer Segmentanteile mit manueller Annotation

In der folgenden Analyse wurden die Annotationen der SBD und der PBD mit der manuellen Annotation verglichen, um die Genauigkeit der beiden Dekodierungsmethoden zu vergleichen.

Dafür wurden jeweils die relativen Anzahlen der acht Segmente zwischen der manuellen Annotation, der SBD und PBD verglichen und in **Abb. 14** dargestellt. Abgebildet sind die relativen Anteile an der Gesamtzahl der Segmente in den undichten Regionen des manuellen Annotationsbereichs.

Bei den Segmenten M und NS war der Anteil von manueller Annotation und PBD mit 17 % und 14 % gleich groß. Der Anteil der SBD lag jeweils zwei Prozent höher. Bei den Segmenten HA und NP war der Anteil von manueller Annotation und PBD mit jeweils sieben Prozent ebenfalls gleich groß. Der Anteil der SBD war im Gegensatz dazu um ein bzw. zwei Prozent niedriger. Bei dem Segment PA waren die Anteile von SBD und PBD mit zwölf Prozent gleich und lagen drei Prozent unter dem Anteil der manuellen Annotation. Die Anteile der SBD und PBD lagen bei PB2 höher als bei der manuellen Annotation, wobei der Anteil der PBD am höchsten war. Bei dem Segment PB1 war der Anteil der PBD mit acht Prozent ebenfalls am höchsten. Die Anteile der manuellen Annotation und SBD waren mit etwa vier Prozent gleich groß. Insgesamt betrug die durchschnittliche Abweichung von manueller Annotation und SBD 1,8 %, die Abweichung der PBD betrug 2,7 %.



**Abb. 14: Vergleich der relativen RNA-Anzahlen, ermittelt durch manuelle Dekodierung, Spot-basierte Dekodierung (SBD) und Pixel-basierte Dekodierung (PBD).** Dargestellt sind die relativen Anteile der acht Influenzasegmente HA, M, NA, NP, NS, PA, PB1 und PB2 an der Gesamtanzahl an detektierten Segmenten in undichten Regionen des manuellen Annotationsbereichs. Die Ergebnisse der manuellen Dekodierung sind in Grün dargestellt, die der SBD in Schwarz und die der PBD in Weiß.

#### 4.1.3.2 Vergleich der Spotzahlen

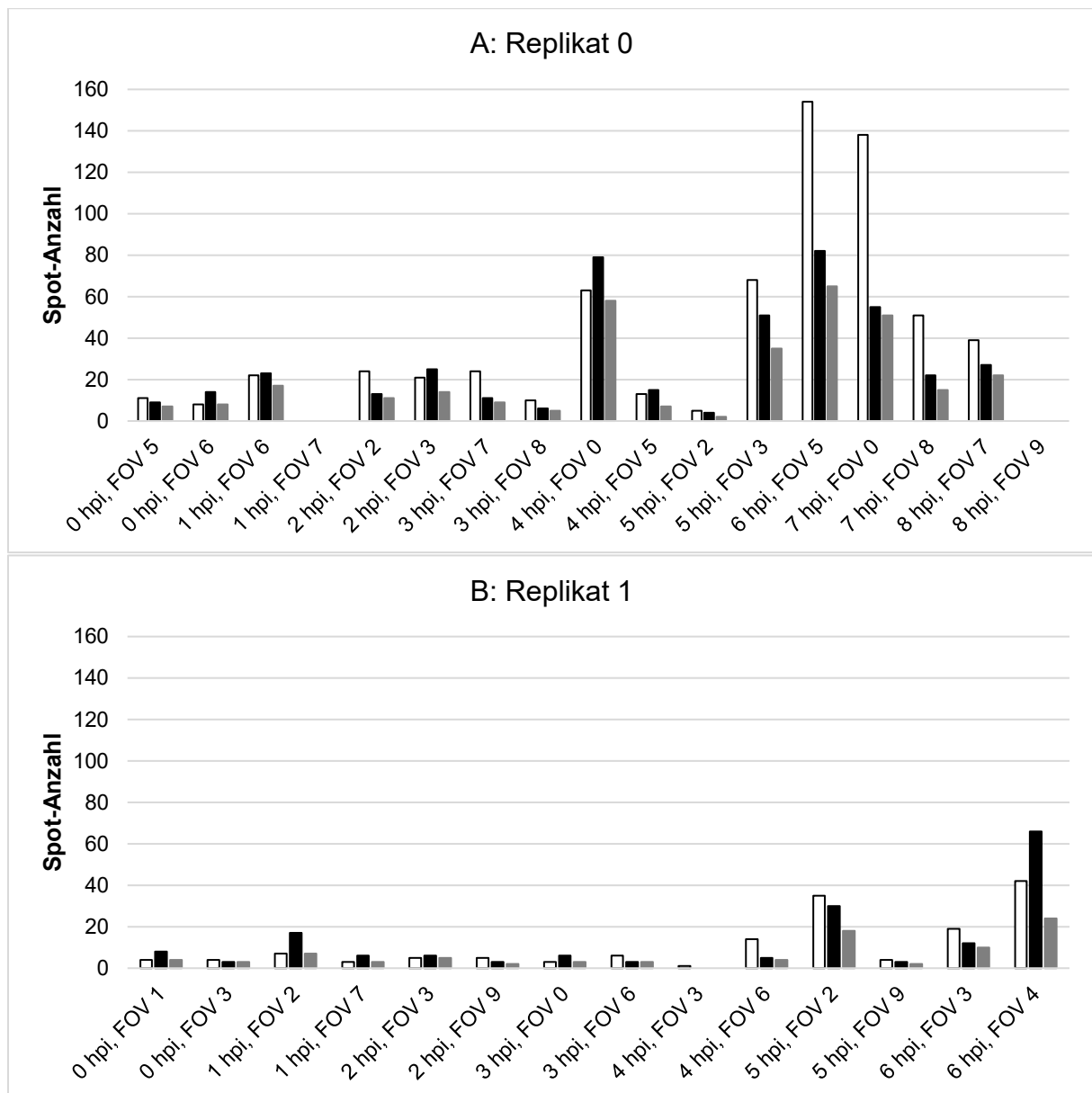
Für eine detailliertere Bewertung der Pixel-basierten Dekodierung wurden die Anzahlen von detektierten Spots pro FOV mit der manuellen Annotation verglichen. Die Analyse wurde so wie der Vergleich von der SBD und manuellen Annotation durchgeführt (siehe 4.1.2.1.1). In **Abb. 15** sind die absoluten Spotanzahlen der manuellen Annotation (weiß) und der Pipelineannotation (schwarz) pro FOV dargestellt. In grau ist die Anzahl der Spots dargestellt, die sowohl manuell als auch von der PBD annotiert wurden. Die Abbildung zeigt nur Spots aus undichten Regionen. Insgesamt detektierte die PBD 1.687 Spots im Annotationsbereich. Davon lagen 728 Spots in undichten Regionen. Die manuelle Annotation detektierte 927 Spots. Jeweils 428 Spots wurden übereinstimmend von PBD und manueller Annotation detektiert. Damit betrug die Anzahl etwa zwei Drittel, verglichen mit der SBD.

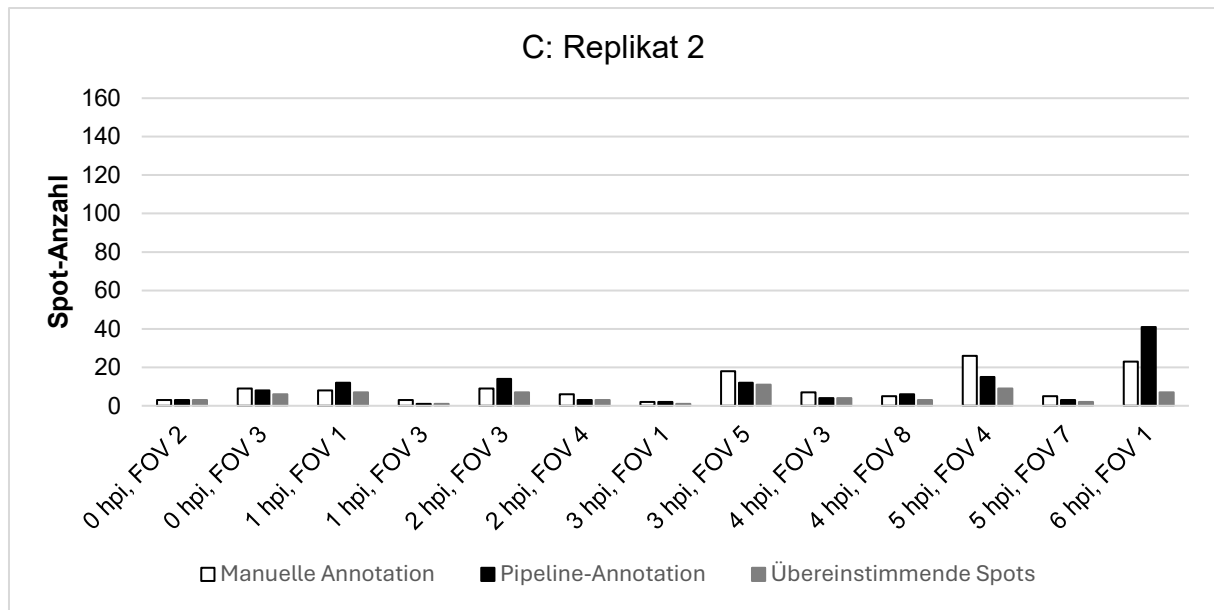
In Abbildung (A) sind die Ergebnisse des Replikats null dargestellt. Zu den Infektionszeitpunkten null bis drei waren sowohl die Anzahlen der manuellen Annotation als auch der Pipelineannotation gering. In „1 hpi, FOV 7“ betrugen beide Anzahlen null. Im Vergleich zur SBD (**Abb. 10**) waren die Ergebnisse dieser frühen Infektionszeiten nahezu identisch. Zu den Infektionszeiten vier bis sechs waren die Spotanzahlen der manuellen Annotation größtenteils höher als die der Pipelineannotation. Vor allem in den FOVs „6 hpi, FOV 5“, „7 hpi, FOV 0“ und „7 hpi, FOV 8“ waren die Anzahlen der manuellen Annotation mindestens doppelt so hoch. Im Vergleich zur SBD zeigte sich, dass die Spotanzahlen zu den späteren Infektionszeitpunkten niedriger waren.

Bei Replikat eins in (B) zeigte sich, dass die Anzahlen der manuellen Annotation zu den Infektionszeitpunkten null bis vier niedrig waren. Sie betrugen einen bis sieben Spots pro FOV. Die Anzahlen der PBD waren ebenfalls niedrig. Lediglich in einem FOV betrug die Anzahl, mit 17 Spots pro FOV, mehr als das doppelte der manuellen Annotation. Im Vergleich zur SBD, bei der die Anzahlen ein Vielfaches der manuellen Annotation betrugen, waren die Anzahlen der PBD deutlich geringer. Zu den Infektionszeitpunkten vier bis sechs waren die Anzahlen der manuellen Annotation, bis auf ein FOV, höher. Lediglich bei „6 hpi, FOV 3“ war die Anzahl der Pipelineannotation höher. Im Vergleich zu der SBD waren die Anzahlen der übereinstimmenden Spots in den meisten FOVs niedriger.

In Abbildung (C) sind die Spotanzahlen der manuellen Annotation und Pipelineannotation des Replikats zwei dargestellt. Die Anzahlen waren in diesem Replikat weitgehend ähnlich. Größere Differenzen zeigten die FOVs „5 hpi, FOV 4“ und „6 hpi, FOV 1“. Dort lag die Anzahl der Pipelineannotation einmal deutlich unter und einmal deutlich über der manuellen Annotation. Im Vergleich zu der SBD waren alle Anzahlen der übereinstimmenden Spots niedriger.

Die Ergebnisse der drei Replikate zeigten im Vergleich zu der SBD also eine deutlich niedrigere Gesamtspotzahl. Zudem waren die Anzahlen der übereinstimmenden Spots in jedem Replikat niedriger.





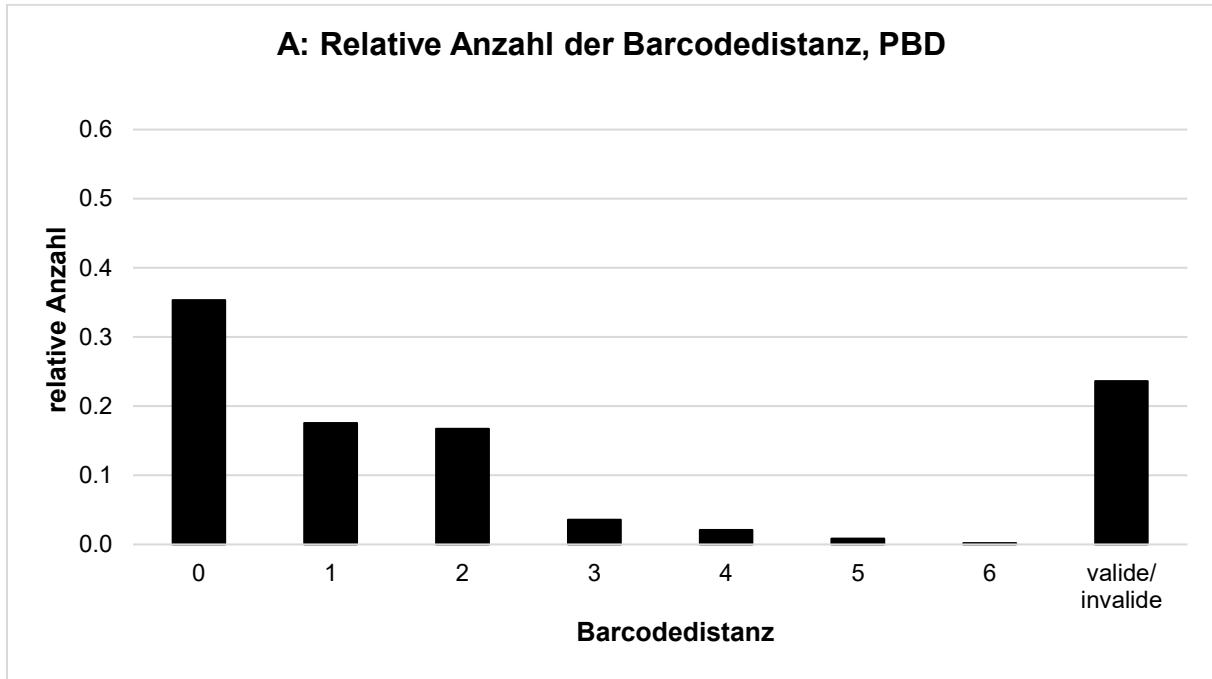
**Abb. 15: Vergleich der Spotanzahlen von manueller Annotation und Pipelineannotation (Pixel-basierte Dekodierung) für drei Replikate.** Die Säulen zeigen die absoluten Spotanzahlen pro FOV. Spots, die ausschließlich manuell annotiert wurden, sind weiß dargestellt, Spots aus der Pipelineannotation schwarz. Spots, die durch beide Methoden detektiert wurden, sind grau dargestellt. Die Abbildung zeigt nur Spots der Pixel-basierten Dekodierung aus undichten Regionen.

#### 4.1.3.3 Barcodedistanz

Für eine genauere Bewertung der PBD wurde die Barcodedistanz zwischen Spots der manuellen Annotation und Spots der PBD berechnet. Die Paarung der insgesamt 428 Spots aus beiden Annotationen und die Berechnung der Barcodedistanz erfolgte dabei wie in 4.1.2.2. Die Ergebnisse sind in **Abb. 16** dargestellt.

**Abb. 16** zeigt, dass Spotpaare mit übereinstimmenden Barcodes, also einer Distanz von null, einen Anteil von 14 % ausmachten. Der Anteil war somit drei Prozent niedriger als bei der SBD (**Abb. 11**). Die Anteile der Distanz eins und zwei waren in der PBD mit etwa 17 % gleich groß. Damit waren sie circa 15 % höher als bei der SBD. Die Anteile der Distanz drei und vier waren in der PBD mit vier und zwei Prozent zwar gering, jedoch betrugen sie etwa das Zehnfache der Anteile der SBD. Die Anteile der Barcodedistanz fünf lagen in der PBD und SBD jeweils unter einem Prozent. In der PBD lag der Anteil der Distanz sechs bei unter einem Prozent und war somit sehr niedrig. In der SBD wurden keine Spotpaare mit einer Distanz von sechs oder höher detektiert. Spotpaare mit einem validen und einem invaliden Spot machten in der PBD einen Anteil von 24 % aus. Im Vergleich dazu war der Anteil in der SBD mit 29 % um fünf Prozent größer. Auffällig zwischen Ergebnissen der SBD und PBD war der Unterschied in den Spotpaaren, die aus zwei invaliden Spots bestanden. Während diese Spotpaare in der SBD ein Anteil von 35 % ausmachten, betrug der Anteil in der PBD null. Die Ursache dafür war,

dass die PBD keine „invalide“-Annotationen vornahm. Insgesamt wurde beobachtet, dass die Anteile an übereinstimmenden Barcodes in der PBD niedriger als in der SBD waren. Zudem entsprachen die Anteile der Distanz eins bis vier in der PBD einem Vielfachen der Anteile der SBD.



**Abb. 16: Barcodedistanzen zwischen Spots der manuellen Annotation und Pipelineannotation (Pixel-basierte Dekodierung).** Insgesamt 478 Spots aus manueller Annotation und Pipelineannotation wurden gepaart und die Barcodedistanz zwischen den Spotpaaren wurde berechnet (siehe 4.1.2.2). Die Säulen zeigen die relativen Spotanzahlen pro Barcodedistanz. Es sind die relativen Anteile der Distanzen null bis sechs und „valide/invalide“ dargestellt. „invalide/valide“ stellt Spotpaare dar, bei denen ein Spot als invalide annotiert wurde und ein Spot als valide. Die Spots wurden mittels Pixel-basierter Dekodierung in undichten Regionen detektiert.

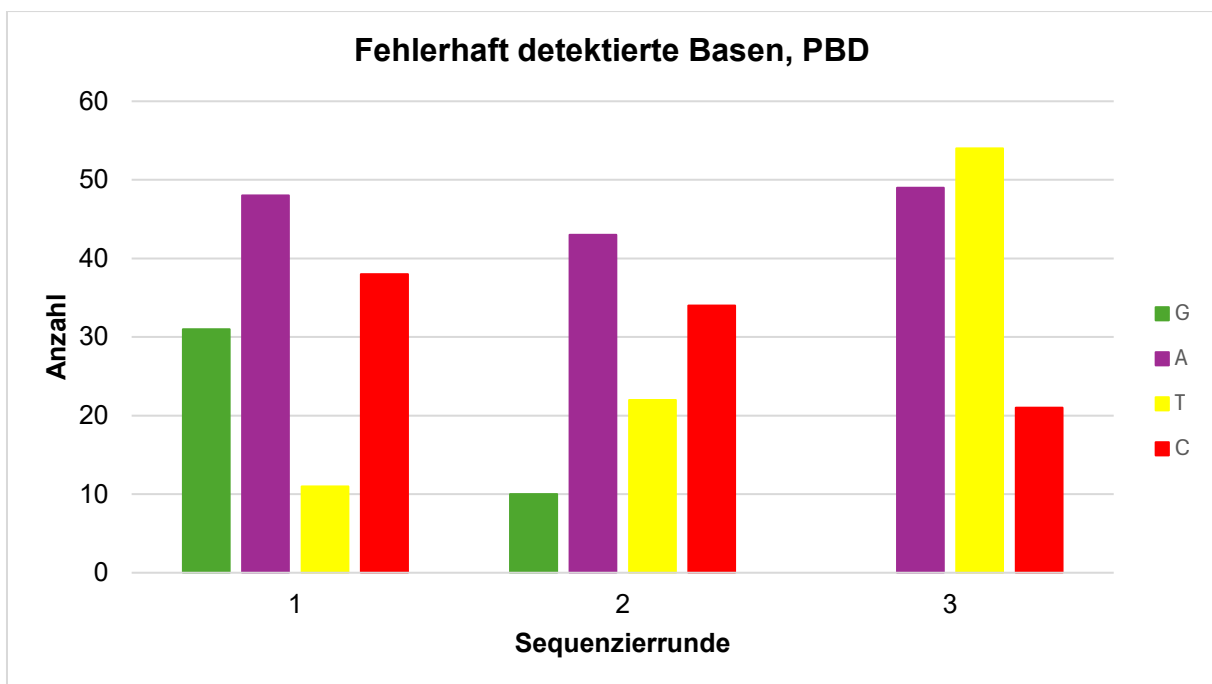
#### 4.1.3.4 Fehlerhafte Detektion von Einzelbasen

In dieser Analyse wurde untersucht, ob es systematische Fehlannotationen der PBD auf Ebene von Einzelbasen gab. Dabei wurde wie bei der Analyse der SBD in 4.1.2.3 vorgegangen. Bei der Analyse der fehlerhaft detektierten Basen wurden die 196 Spotpaare mit einer Barcodedistanz zwischen eins und sechs aus **Abb. 16** betrachtet.

Die **Abb. 17** zeigt, dass in der ersten Runde die Base Adenin am häufigsten falsch detektiert wurde. Die Anzahlen der Basen Cytosin und Guanin lagen mit 38 und 31 falsch detektierten Basen dahinter. Die Base Thymin wurde dagegen nur elfmal falsch detektiert. In der zweiten Sequenzerrunde wurden ebenfalls die Basen Adenin und Cytosin am häufigsten falsch detektiert, wobei beide Anteile etwas niedriger als in der ersten Runde waren. Im Gegensatz

zur ersten Runde war die Anzahl der Falschdetektion bei Guanin am geringsten. In der dritten Runde waren die Anzahlen von Thymin und Adenin am größten. Sie betrugen 54 und 49. Die Base Cytosin wurde 21-mal falsch detektiert. Das stellte den niedrigsten Wert der drei Runden dar. Die Base Guanin wurde in der dritten Runde gar nicht falsch detektiert.

Im Vergleich zu der SBD (**Abb. 12**) unterschieden sich die Ergebnisse stark. Während die Base Adenin in der SBD weniger falsch detektiert wurde, wurde sie in der PBD am häufigsten falsch detektiert. Außerdem sank die Anzahl der falsch detektierten Cytosinbasen in der PBD mit jeder Runde, während sie in der SBD mit jeder Runde zunahm. Die Anzahlen der falsch detektierten Thyminbasen stiegen in der PBD mit jeder Runde. In der SBD dagegen war die Anzahl über alle Runden hoch. Insgesamt waren die Anzahlen der falsch detektierten Basen auf einem höheren Niveau, wobei auch die Anzahl der analysierten Spotpaare in der PBD höher war.



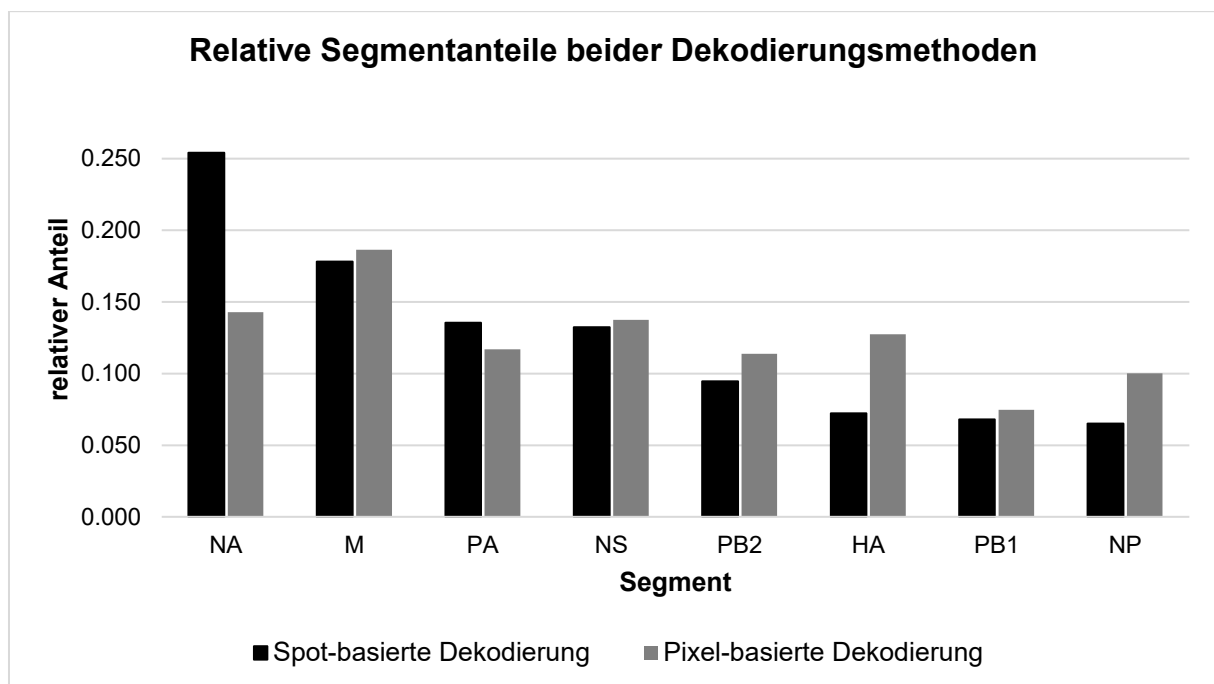
**Abb. 17: Anzahl an fehlerhaft annotierten Basen durch Pixel-basierte Dekodierung in undichten Regionen.** Für die insgesamt 196 gepaarten Spots mit einer Barcodedistanz von eins bis sechs aus **Abb. 16** wurden die Anzahlen der fehlerhaft detektierten Basen pro Sequenzerrunde bestimmt. Guanin ist in Grün dargestellt, Adenin in Violett, Thymin in Gelb und Cytosin in Rot.

#### 4.1.3.5 Relative Segmentanteile beider Dekodierungsmethoden

Da die Dekodierung von Spots in dichten Regionen mittels SBD inkonsistent war (**Abb. 13 A**), wurde die Annotation von Spots in dichten Regionen mittels PBD analysiert. Die Methode beruht nicht auf der Detektion ganzer Spots, sondern dekodiert auf Pixelebene. Die Einzelpixel werden dann im Anschluss zu Spots gruppiert. Daher wurde angenommen, dass die PBD

besser für dichte Regionen geeignet ist. Eine Validierung der PBD in den dichten Regionen durch den Vergleich mit einer manuellen Annotation war nicht möglich, da die Unterscheidung einzelner Spots in zu dichten Regionen nicht möglich war. Daher wurden im Folgenden die relativen Anteile der Segmente von der SBD in undichten Regionen und der PBD in dichten Regionen verglichen.

Die **Abb. 18** zeigt die relativen Anteile der Segmente an der Gesamtsegmentanzahl aller Replikate, Infektionszeitpunkte und FOVs. Die Ergebnisse der SBD sind in Schwarz dargestellt, die Anteile der PBD in Grau. Das Segment NA machte in der SBD einen Anteil von circa 25 % aus. In der PBD war der Anteil mit etwa 14 % deutlich geringer. Im Gegensatz dazu waren die Anteile der Segmente HA und NP in der PBD um sechs und drei Prozent höher. Der Anteil von PA betrug 14 % in der SBD und war damit zwei Prozent höher als bei der PBD. Bei dem Vergleich der Anteile des Segments PB2 zeigte sich das Gegenteil. Dort war der Anteil der PBD zwei Prozent höher als der Anteil der SBD. Bei den restlichen Segmenten betrugen die Unterschiede zwischen den beiden Methoden maximal ein Prozent. Insgesamt waren die Unterschiede bei NA, HA und NP am größten. Bei NA lag der Anteil der SBD deutlich höher und bei HA und NP war der Anteil der PBD höher. Die Anteile der restlichen Segmente zeigten nur geringe Unterschiede.



**Abb. 18: Vergleich der relativen Segmentanzahlen, ermittelt durch Spot-basierte Dekodierung (SBD) und Pixel-basierte Dekodierung (PBD).** Dargestellt sind die relativen Anteile der acht vRNP-Segmente NA, M, PA, NS, PB2, HA, PB1 und NP an der Gesamtanzahl detektierter Segmente aller Replikate, Infektionszeiten und FOVs. Die Ergebnisse der SBD sind in Schwarz dargestellt, von der PBD in Grau.



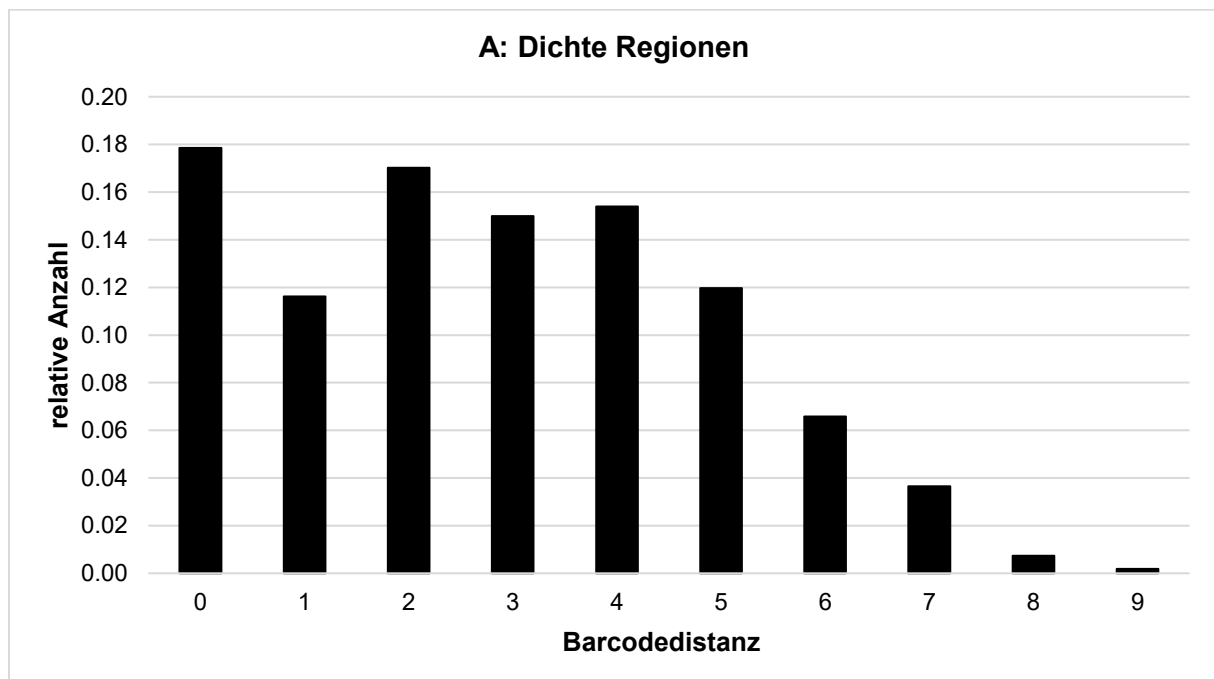
#### 4.1.3.6 Konsistenz in überlappenden FOV-Bereichen

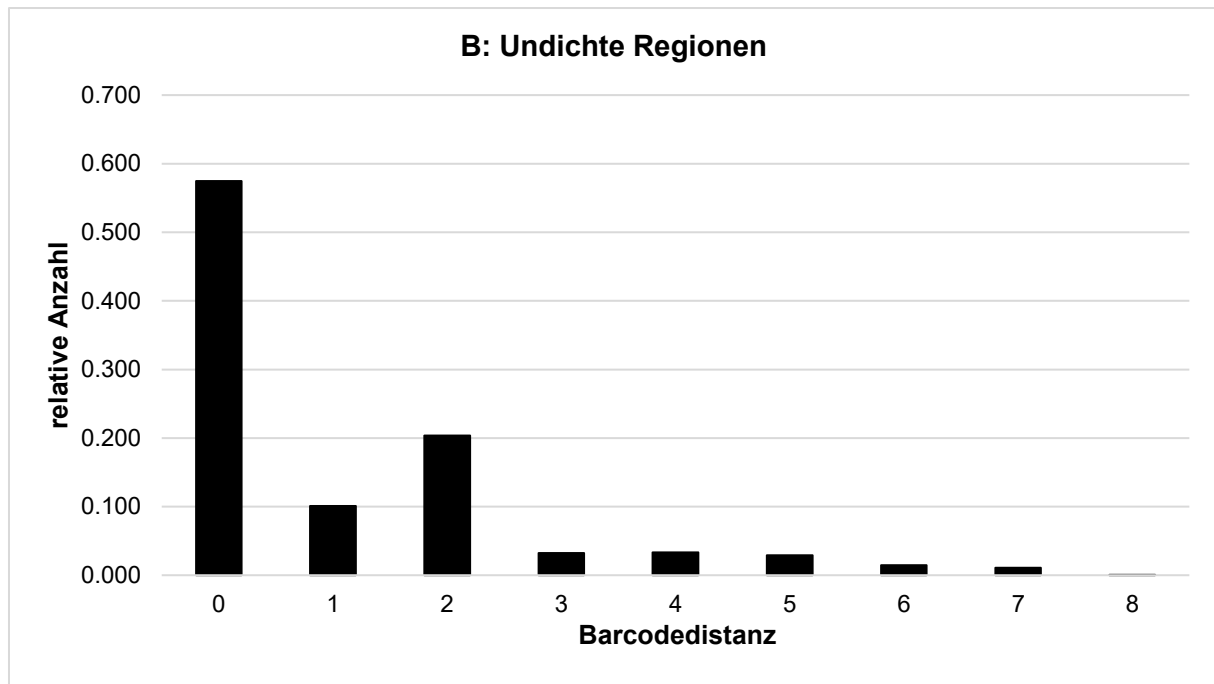
Für eine detailliertere Bewertung der PBD wurde ebenfalls die Konsistenz der Annotation analysiert. Dabei wurden Spots aus überlappenden FOV-Bereichen gepaart und die Barcodedistanzen der Spotpaare wurden verglichen. Dabei wurde wie bei der Analyse der SBD in 4.1.2.4 vorgegangen. In **Abb. 19** sind die Ergebnisse aus den dichten und undichten Regionen abgebildet.

In (A) sind die relativen Anteile der Barcodedistanzen der gepaarten Spots aus den dichten Regionen dargestellt. Insgesamt wurden 185.305 Spots in dieser Region detektiert, wovon 43.782 Spots gepaart wurden. Die Abbildung zeigt die relativen Anteile der Spotpaare mit Barcodedistanzen zwischen null und neun. Am höchsten war der Anteil an Spotpaaren mit übereinstimmenden Spots (Distanz null), er betrug 18 %. Der Anteil von Spotpaaren mit einer Distanz von zwei war mit 17 % etwa gleich groß. Paare mit den Barcodedistanzen drei und vier machten jeweils einen Anteil von 15 % aus. Im Vergleich dazu waren die Anteile der Spotpaare mit den Distanzen eins und fünf um jeweils drei Prozent niedriger (12 %). Deutlich geringer waren die Anteile der Barcodedistanzen sechs bis neun. Während die Anteile der Distanzen sechs und sieben, circa sieben bzw. vier Prozent betrugen, lagen die Anteile der Distanzen acht und neun bei einem Prozent und weniger. Zusammenfassend zeigte sich, dass sich der Anteil von Spotpaaren mit übereinstimmenden Spots nicht deutlich von Spotpaaren mit Distanzen von eins bis fünf unterschied.

In (B) sind die Anteile der Barcodedistanzen der gepaarten Spots aus den undichten Regionen dargestellt. Die Gesamtspotzahl der undichten Region betrug 40.942. Davon konnten 5.350 Spots gepaart werden. Mit Abstand den größten Anteil machten übereinstimmende Spotpaare aus. Dieser lag bei 57 % und betrug somit mehr als das Dreifache, verglichen mit den dichten Regionen (A). Der Anteil der Distanz zwei war mit 20 % am zweitgrößten. Das war vergleichbar mit dem Anteil aus den dichten Regionen (17 %). Ebenfalls vergleichbar mit den dichten Regionen war der Anteil der Distanz eins. In den dichten Regionen betrug er zwölf Prozent und in den undichten Regionen zehn Prozent. Die Spotpaare mit den Distanzen drei, vier und fünf hatten jeweils einen Anteil von drei Prozent. Im Vergleich zu den dichten Regionen waren diese Anteile drei- bis viermal kleiner. Mit jeweils einem Prozent waren die Anteile von Spotpaaren mit einer Distanz von sechs und sieben gleich groß. Damit betrugen sie nur ein Bruchteil der Anteile aus den dichten Regionen. Der Anteil der Distanz acht war mit 0,1 % sehr gering. Spotpaare mit einer Barcodedistanz von neun wurden, im Gegensatz zu den dichten Regionen, nicht detektiert. Die Ergebnisse der undichten Region zeigten somit, dass der Anteil an Spotpaaren mit übereinstimmenden Spots, im Vergleich zu den dichten Regionen, deutlich höher war. Zudem waren die Anteile von nicht übereinstimmenden Spots insgesamt deutlich niedriger.

Der Vergleich der relativen Anteile der Barcodedistanzen aus dichten und undichten Regionen zeigte, dass die PBD in undichten Regionen konsistenter war. Dies zeigte sich auch bei der SBD (**Abb. 13**). Im Vergleich von SBD und PBD zeigte sich, dass in den dichten Regionen die Anteile von Spotpaaren mit übereinstimmenden Spots und nicht-übereinstimmenden Spots keine wesentlichen Unterschiede zeigten. In den dichten Regionen wurde beobachtet, dass die relativen Anzahlen von übereinstimmenden Spots in der PBD höher als in der SBD waren. Bei dem Vergleich der absoluten Anzahlen betrug die Anzahl in der SBD 6.853 und in der PBD 3.074. Der Vergleich der absoluten Anzahlen zeigte somit, dass die Anzahl an übereinstimmenden Spots in der SBD doppelt so hoch wie in der PBD war. Außerdem waren, gemessen in absoluten Zahlen, alle Anzahlen der Distanzen eins bis sieben in der PBD höher als in der SBD. Die PBD zeigte also mehr Spotpaare mit nicht-übereinstimmenden Spots, obwohl die Gesamtzahl der Spotpaare niedriger war.





**Abb. 19: Barcodedistanzen zwischen Spotpaaren aus überlappenden FOV-Bereichen (Pixel-basierte Dekodierung).** Überlappende FOV-Bereiche wurden ausgeschnitten, registriert und die Spots beider FOVs, deren euklidische Distanz weniger als fünf Pixel betrug, wurden gepaart. Berechnung der Barcodedistanz erfolgte wie in 4.1.2.4. Beide Abbildungen zeigen die relativen Anteile der Spotpaare mit einer bestimmten Barcodedistanz. (A) Dargestellt sind insgesamt 185.305 Spotpaare aus dichten Regionen. (B) Dargestellt sind 5.350 Spotpaare aus undichten Regionen.

## 4.2 Analyse der Genom Assemblierung

Da die Ergebnisse der Evaluation der Pipeline zeigten, dass die konsistenteste und akkurateste Dekodierung der Spots durch die SBD in undichten Regionen erzielt wurde, wurden nur diese Spots für die nachfolgenden Analysen betrachtet. In den folgenden Analysen wurden die detektierten Segmente und Komplexe genauer analysiert, um mögliche Assemblierungswege des Influenzavirus zu postulieren. Dabei wurden Anzahlen von detektierten Komplexen verglichen und mit berechneten Komplexwahrscheinlichkeiten gegenübergestellt. Außerdem wurde die Verpackung der Segmente zu einem vollständigen Viruspartikel in einer monomeren Assemblierung modelliert.

### 4.2.1 Relative Anzahlen der Komplexe

Im Folgenden wurden die relativen Anteile der Einzelkomplexe innerhalb der jeweiligen Komplexgröße berechnet. Für jeden Einzelkomplex wurde das arithmetische Mittel der insgesamt 15 Messwerte aus den jeweils fünf Replikaten der Infektionszeitpunkte vier bis

sechs berechnet. In **Abb. 20** sind jeweils die acht häufigsten Komplexe einer Komplexgröße dargestellt. Die restlichen Ergebnisse befinden sich im Anhang (**Abb. 22 - Abb. 26**). Das arithmetische Mittel ist jeweils als großer weißer Punkt dargestellt. Die 15 Einzelmesswerte pro Komplex sind als kleine schwarze Punkte dargestellt. Zusätzlich ist jeweils rechts in der Abbildung die Summe der Einzelwerte von jedem Komplex abgebildet. Diese repräsentiert den relativen Anteil des Komplexes an allen Komplexen der gleichen Größe. Da vor allem bei größeren Komplexen in einigen Replikaten keine Komplexe detektiert wurden, kommt es zur Überlagerung von Einzelmesswerten an der Nulllinie.

In (A) sind die relativen Anzahlen der acht Einzelsegmente abgebildet. Die Gesamtzahl der Einzelsegmente betrug 181.016. Die Abbildung zeigt, dass PB2 am häufigsten detektiert wurde, wobei ein Drittel der Messwerte über dem arithmetischen Mittel lag. Insgesamt stellten die PB2-Segmente einen Anteil von 28 % dar. Die Segmente PB1 und PA hatten einen Gesamtanteil von 21 % und 19 %. Bei beiden Segmenten lag ein überwiegender Teil der Messwerte über dem Mittelwert. Die Gesamtanteile der Segmente NS, NP, NA, M und HA lagen zwischen vier und zehn Prozent. Die Einzelwerte waren dabei dichter um den Mittelwert verteilt.

In Abbildung (B) sind die relativen Anteile der acht häufigsten Zweierkomplexe dargestellt. Die Gesamtzahl der Zweierkomplexe betrug 68.479. Der Komplex PB2,PB1 machte einen Anteil von 14 % aller 28 möglichen Zweierkomplexe aus. Auffällig dabei war, dass der Anteil eines Replikats (Replikat 0, sechs Stunden nach Infektion) bereits über zehn Prozent der gesamten Zweierkomplexe ausmachte. Bei den relativen Anzahlen der Einzelkomplexe (A) konnte ebenfalls beobachtet werden, dass dieses Replikat bereits einen großen Anteil der PB1-Komplexe ausmachte. Mit neun und acht Prozent waren die Gesamtanteile von PB2,PA und PB2,NS ebenfalls hoch. Die Gesamtanteile der restlichen Komplexe lagen zwischen drei und sechs Prozent. Die sieben häufigsten Zweierkomplexe stellten die sieben möglichen Komplexe dar, die PB2 beinhalten.

Abbildung (C) zeigt die acht häufigsten der insgesamt 56 verschiedenen Dreierkomplexe. Insgesamt wurden 61.361 Komplexe detektiert. Mit einem Gesamtanteil von sechs Prozent war der Komplex PB2,PB1,PA am häufigsten. Den zweitgrößten Gesamtanteil hatte der Komplex PB2,PB1,NS mit fünf Prozent. Die nächsten vier häufigsten Komplexe machten jeweils einen Gesamtanteil von circa vier Prozent aus. Bei den meisten Komplexen war die Verteilung der Einzelmesswerte um den Mittelwert gleichmäßig, wobei die Streuung der Werte relativ groß war. Bei den Komplexen PB2,PB1,M und PB2,PA,NP lagen jeweils nur zwei bzw. ein Messwert über dem Mittelwert. Die sechs häufigsten Komplexe enthielten die Segmente PB2 und PB1 und stellten somit alle möglichen Dreierkomplexe mit diesen beiden Segmenten

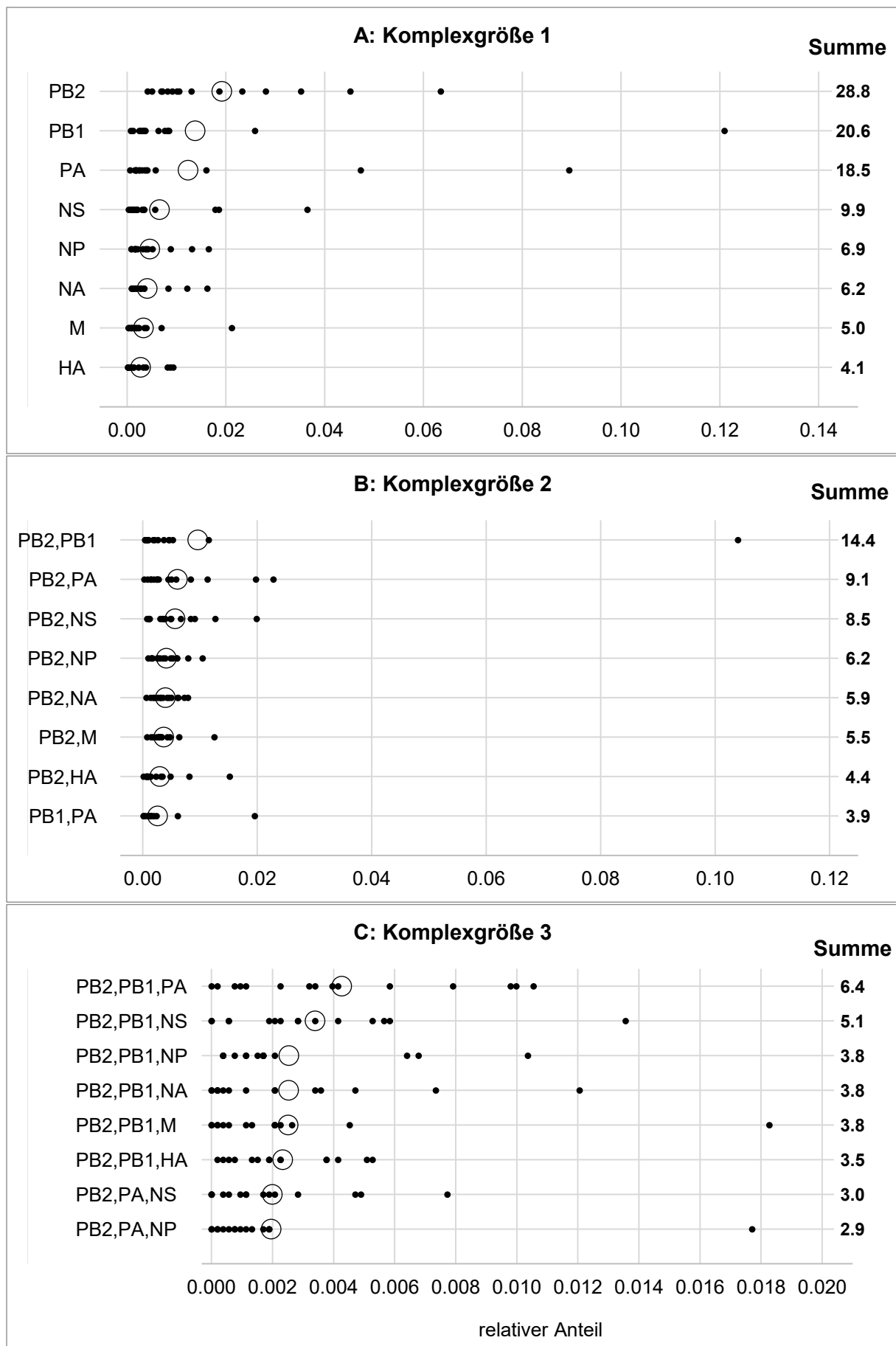
dar. Die nächsten fünf häufigsten Dreierkomplexe enthielten jeweils die Segmente PB2 und PA. Die Ergebnisse sind im Anhang dargestellt (**Abb. 23**).

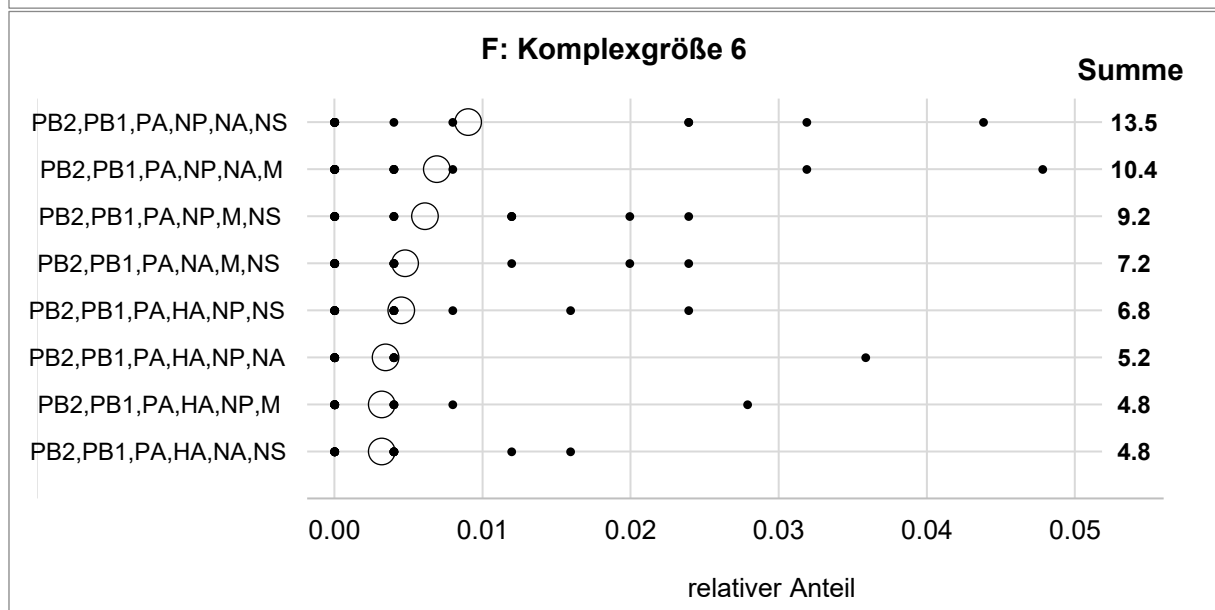
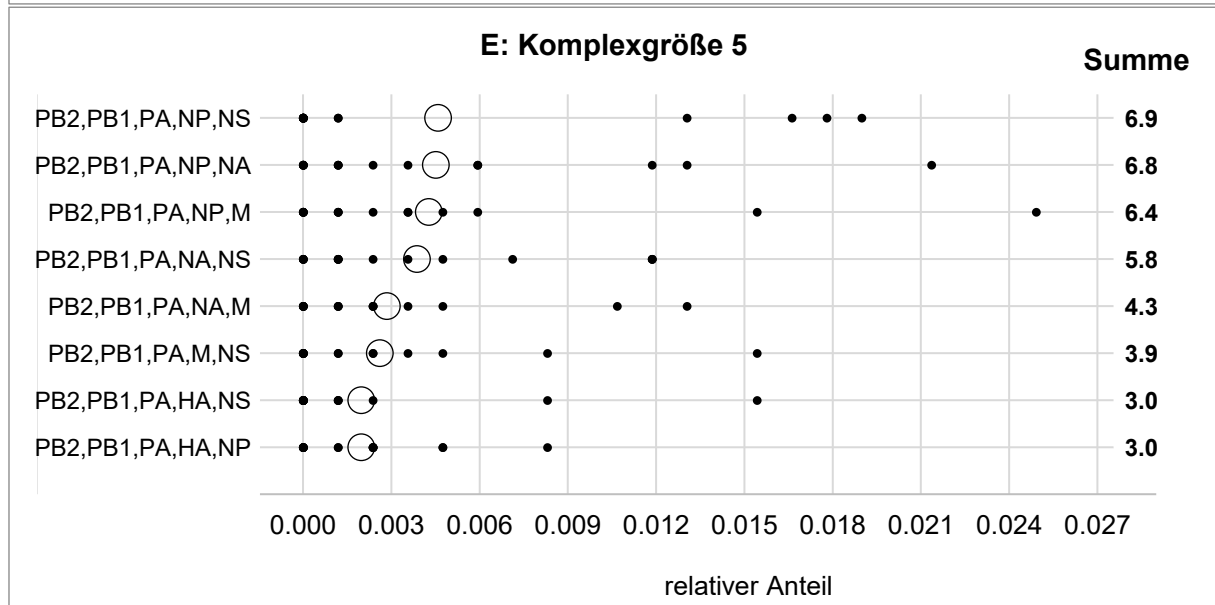
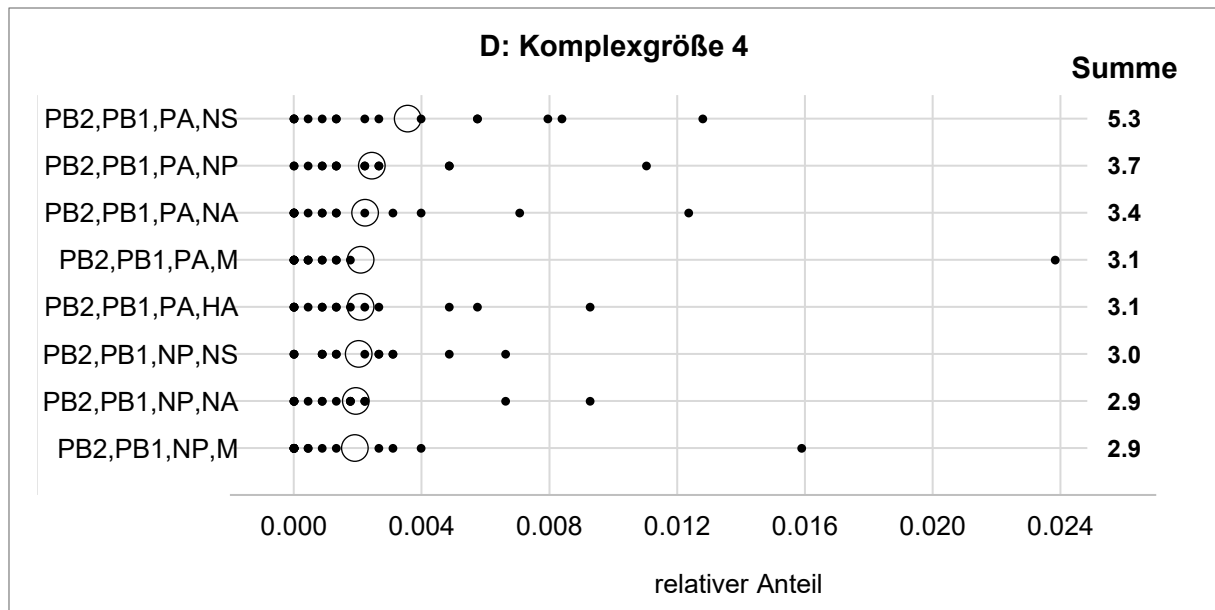
Abbildung (D) zeigt die relativen Anteile der acht abundantesten Komplexe der insgesamt 54.962 Viererkomplexe. Es wurden alle 70 einzigartige Komplexe detektiert. Mit fünf Prozent hatte der Komplex PB2,PB1,PA,NS den größten Gesamtanteil. Die Gesamtanteile von PB2,PB1,PA,NP und PB2,PB1,PA,NA waren mit 3,7 % und 3,4 % am zweit- und dritthöchsten. Die Gesamtanteile der restlichen dargestellten Komplexe betrug circa drei Prozent. Die Verteilung der Einzelwerte um die Mittelwerte war bei fast allen Komplexen relativ gleichmäßig. Lediglich PB2,PB1,PA,M zeigte einen starken Ausreißer, sodass nur ein Wert oberhalb des Mittelwerts lag. Die fünf häufigsten Komplexe stellten alle Möglichkeiten von Viererkomplexen mit PB2,PB1 und PA dar.

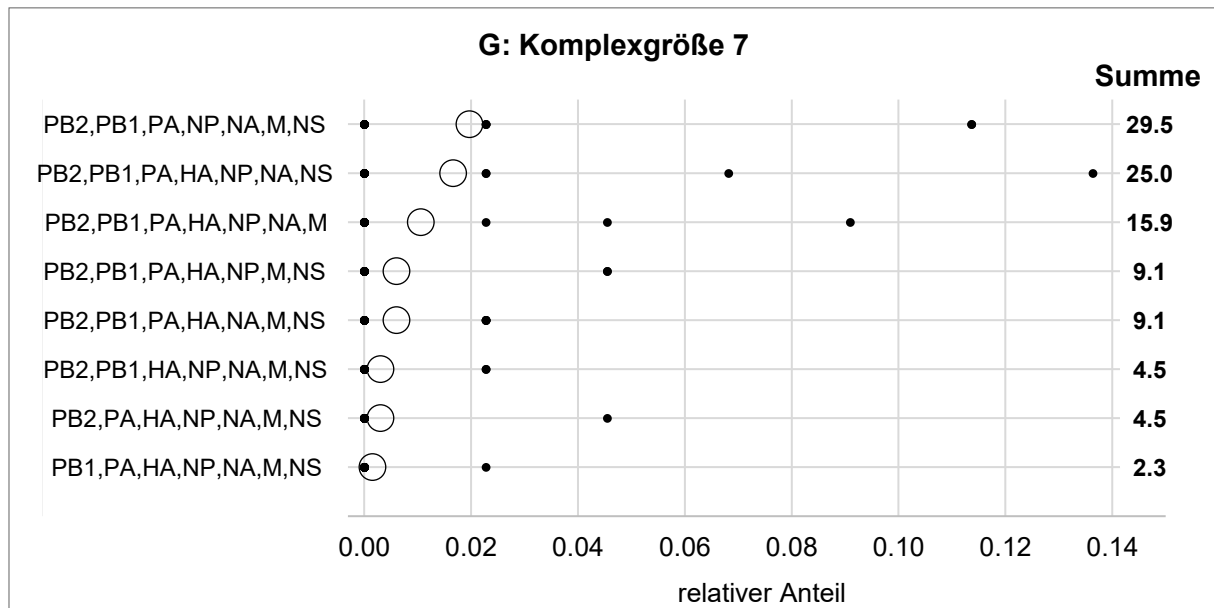
Abbildung (E) zeigt die acht häufigsten Fünferkomplexe. Alle der 56 möglichen einzigartigen Komplexe wurden detektiert. Die Gesamtanzahl der Komplexe betrug 36.673. Die drei häufigsten Komplexe beinhalteten die vier Segmente PB2, PB1, PA und NP und machten jeweils einen Gesamtanteil von 6,4 % bis 6,8 % aus. Die häufigsten zehn Komplexe beinhalteten jeweils die Segmente PB2, PB1 und PA. Die relativen Anzahlen der restlichen Komplexe sind im Anhang dargestellt (**Abb. 25**). Die Einzelwerte waren ungleichmäßig und breit um den Mittelwert verteilt. Viele Einzelwerte lagen bei null und jeweils zwei bis vier Werte lagen über dem Mittelwert.

Die relativen Anteile der Sechserkomplexe sind in (F) dargestellt. Von den 28 möglichen Komplexen wurden 27 detektiert. Die Gesamtanzahl betrug 18.309. Der Komplex PB2,PB1,PA,NP,NA,NS wurde am häufigsten detektiert und machte einen Gesamtanteil von 13,5 % aus. Alle dargestellten Komplexe beinhalteten die Segmente PB2, PB1 und PA. Außerdem beinhalteten sechs dieser Komplexe zusätzlich das Segment NP. Auffällig war auch die hohe Abundanz von Komplexen, die M und NS enthielten. Die Verteilung der Einzelwerte zeigte, dass jeder Komplex in maximal vier der 15 Replikate detektiert wurde.

Die relativen Anteile der acht einzigartigen Siebenerkomplexe sind in (G) dargestellt. Die Gesamtzahl der detektierten Komplexe betrug 6.048. Der Komplex, der am häufigsten detektiert wurde, war PB2,PB1,PA,NP,NA,M,NS. Sein Gesamtanteil betrug 30 %. Der Komplex, der am zweitmeisten detektiert wurde, hatte einen Gesamtanteil von 25 %. In diesem Komplex fehlte das Segment M. In dem dritthäufigsten Komplex fehlte das Segment NS und der Gesamtanteil betrug 16 %. Die restlichen fünf Komplexe hatten einen Anteil von jeweils weniger als zehn Prozent. Die Verteilung der Einzelwerte zeigte, dass jeder Komplex in maximal drei Replikaten detektiert wurde.







**Abb. 20: Relative Komplexzahlen pro Komplexgröße der Infektionszeitpunkte vier bis sechs.** (A) stellt die relativen Häufigkeiten der Komplexe der Größe eins dar, (B) die der Größe zwei, (C) die der Größe drei, (D) die der Größe vier, (E) die der Größe fünf, (F) die der Größe sechs und (G) die der Größe sieben. Es sind jeweils die acht häufigsten Komplexe dargestellt. Die restlichen Komplexanzahlen befinden sich im Anhang (Abb. 22 - Abb. 26). Die kleinen schwarzen Punkte zeigen die relativen Häufigkeiten der jeweils fünf Replikate der Infektionszeitpunkte vier bis sechs ( $n=15$ ) innerhalb der jeweiligen Komplexgröße. Die großen weißen Punkte zeigen das arithmetische Mittel. Die Summe rechts beschreibt die relative Anzahl des Komplexes an der Gesamtzahl aller Komplexe derselben Größe. Durch Überlagerung von Einzelwerten an der Nulllinie, sind vor allem bei den größeren Komplexen nicht alle Einzelwerte erkennbar. Alle Komplexe wurden mittels Spot-basierter Dekodierung in undichten Regionen detektiert.

#### 4.2.2 Abundanz und Wahrscheinlichkeiten der Komplexe

In der folgenden Analyse wurde die Wahrscheinlichkeit für das Auftreten aller einzigartigen Komplexe mit den tatsächlich detektierten Abundanzen der Komplexe verglichen. So konnte für jeden Komplex gezeigt werden, ob die detektierte Abundanz durch die rein zufällige Zusammenlagerung der Einzelsegmente erklärt werden kann. Durch das Multiplizieren der relativen Abundanzen der Einzelsegmente, konnte für jeden Komplex die Wahrscheinlichkeit für eine zufällige Assemblierung berechnet werden. In der folgenden Abbildung (Abb. 21) sind die berechneten Wahrscheinlichkeiten und beobachteten Abundanzen im Vergleich dargestellt. Die Säulen zeigen jeweils das arithmetische Mittel von den Infektionszeitpunkten vier bis sechs mit jeweils fünf Replikaten ( $n = 15$ ). Alle Einzelsegmente und Komplexe wurden mittels Spot-basierter Dekodierung in undichten Regionen detektiert. Bei den Komplexen der Größen zwei bis sechs sind links der Trennlinie jeweils die sechs höchsten relativen



Abundanzen und die entsprechenden Komplexwahrscheinlichkeiten dargestellt. Rechts der Trennlinie sind jeweils die niedrigsten relativen Abundanzen und die dazugehörigen Wahrscheinlichkeiten dargestellt. Die Ergebnisse der restlichen Komplexe befinden sich im Anhang (**Abb. 27 - Abb. 31**). Die Fehlerbalken zeigen den Standardfehler. Die Sternchen zeigen die Signifikanz ( $\alpha = 0,05$ ) für die Abweichung der Wahrscheinlichkeit zur relativen Abundanz. Sie wurde durch einen unabhängigen Zweistichproben T-Test ermittelt. Die berechneten p-Werte wurden nach der Methode von Bonferroni für multiples Testen korrigiert. (Bland und Altman 1995). Ein Sternchen markiert einen p-Wert kleiner gleich 0,05. Zwei Sternchen markieren einen p-Wert kleiner gleich 0,01 und drei Sternchen einen p-Wert kleiner gleich 0,001.

In der Abbildung (A) sind die relativen Abundanzen und Komplexwahrscheinlichkeiten von insgesamt zwölf Zweierkomplexen im Vergleich dargestellt. Mit circa zehn Prozent waren die relativen Abundanzen der Komplexe PA,NA und PB2,PA am höchsten. Dabei zeigte sich vor allem bei PB2,PA eine deutlich höhere Abundanz des Komplexes im Vergleich zu der berechneten Wahrscheinlichkeit. Die relativen Abundanzen von NP,NA und PB1,PA waren mit etwa neun Prozent etwas geringer. Dabei war die relative Abundanz von PB1,PA signifikant höher als aufgrund der Komplexwahrscheinlichkeit zu erwarten gewesen wäre. Der Unterschied zwischen beobachteter und erwarteter Abundanz war bei M,NS ebenfalls hoch, jedoch nicht signifikant. Bei dem Komplex HA,NA wurden keine großen Unterschiede zwischen der relativen Abundanz und Komplexwahrscheinlichkeit beobachtet. Bis auf PB2,HA waren die sechs Komplexe mit den niedrigsten relativen Abundanzen deutlich weniger abundant als durch die Komplexwahrscheinlichkeit zu erwarten gewesen wäre. Von den sechs Komplexen mit der niedrigsten Abundanz beinhalteten vier Komplexe das Segment PB1. Außerdem war die relative Abundanz von PB1,NS signifikant niedriger als die Wahrscheinlichkeit.

Abbildung (B) zeigt die relativen Abundanzen und Komplexwahrscheinlichkeiten der sechs abundantesten und sechs seltensten Dreierkomplexe. Die beiden Komplexe mit der höchsten Abundanz waren PB2,PA,NA und PB2,PB1,PA. Die relativen Abundanzen betrugen 6,5 % und 5,7 %. Weitere abundante Komplexe waren HA,NP,NA und PB1,PA,NA mit 4,9 % und 4,4 %. Die relativen Abundanzen von PB2,PA,M und NP,NA,NS waren mit 3,3 % gleich. Die größte Differenz zwischen Abundanz und Wahrscheinlichkeit zeigte der Komplex PB2,PB1,PA. Unter den sechs abundantesten Komplexen beinhalteten vier Komplexe mindestens eins der Segmente PB2, PB1 und PA. Die relativen Abundanzen der seltenen Komplexe waren mit 0,4 % bis 0,6 % fast gleich. Bei allen dieser Komplexe lag die berechnete Wahrscheinlichkeit über der relativen Abundanz. Die Differenz war bei den Komplexen PB1,HA,NA und PB1,NA,M am größten und bei PB1,NA,M hoch signifikant. Fünf der sechs seltenen Komplexe beinhalteten das Segment PB1 kombiniert mit M oder NS.

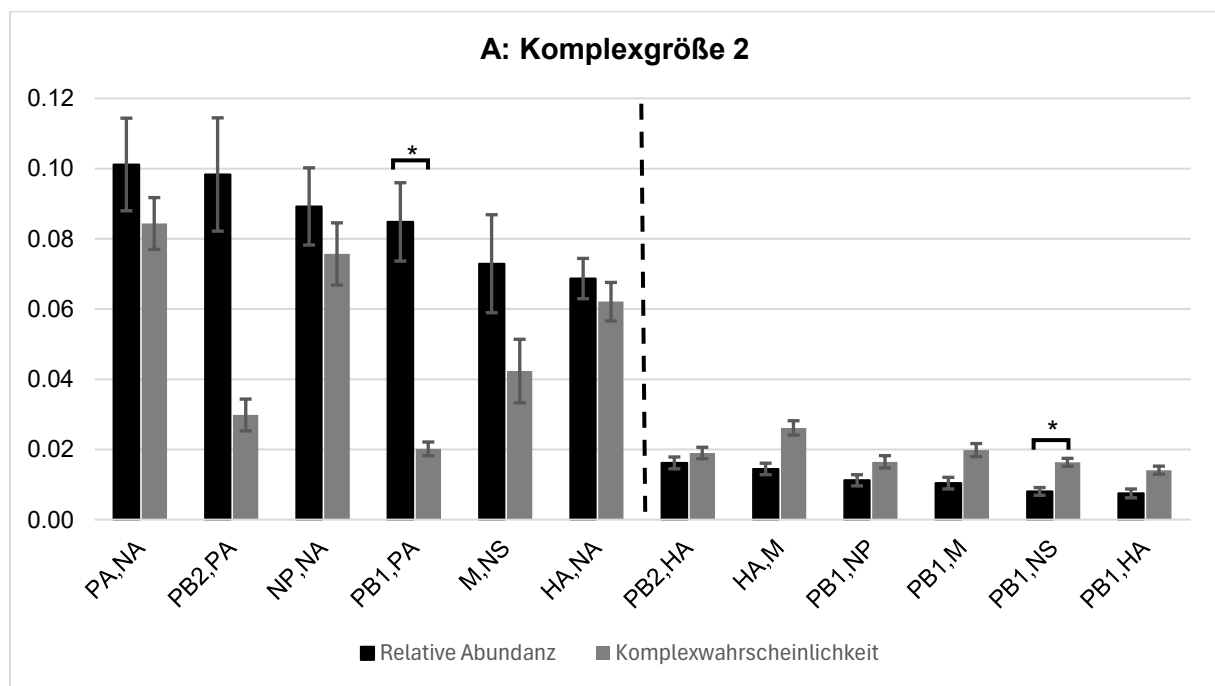
Die Abbildung (C) zeigt die relativen Abundanzen und Komplexwahrscheinlichkeiten der Viererkomplexe im Vergleich. Der abundanteste Komplex war PB2,PB1,PA,NA. Dieser machte 5,8 % der gesamten Viererkomplexe aus. Der Komplex HA,NA,M,NS wurde am zweithäufigsten detektiert und machte einen Anteil von 4,2 % aus. Die anderen vier abundantesten Komplexe machten jeweils einen Anteil zwischen 3,0 % und 3,5 % aus. Bei allen sechs abundanten Komplexen war die relative Abundanz höher als die Wahrscheinlichkeit. Am größten war die Differenz bei den Komplexen PB2,PB1,PA,NA und PB2,PB1,PA,M. Die relative Abundanz der am wenigsten detektierten Komplexe betrug zwischen 0,2 % und 0,4 %. Außerdem war die berechnete Wahrscheinlichkeit höher als die relative Abundanz. Signifikant waren diese Differenzen bei den Komplexen PB2,PB1,NA,NS; PB1,HA,NA,NS und PA,NP,M,NS.

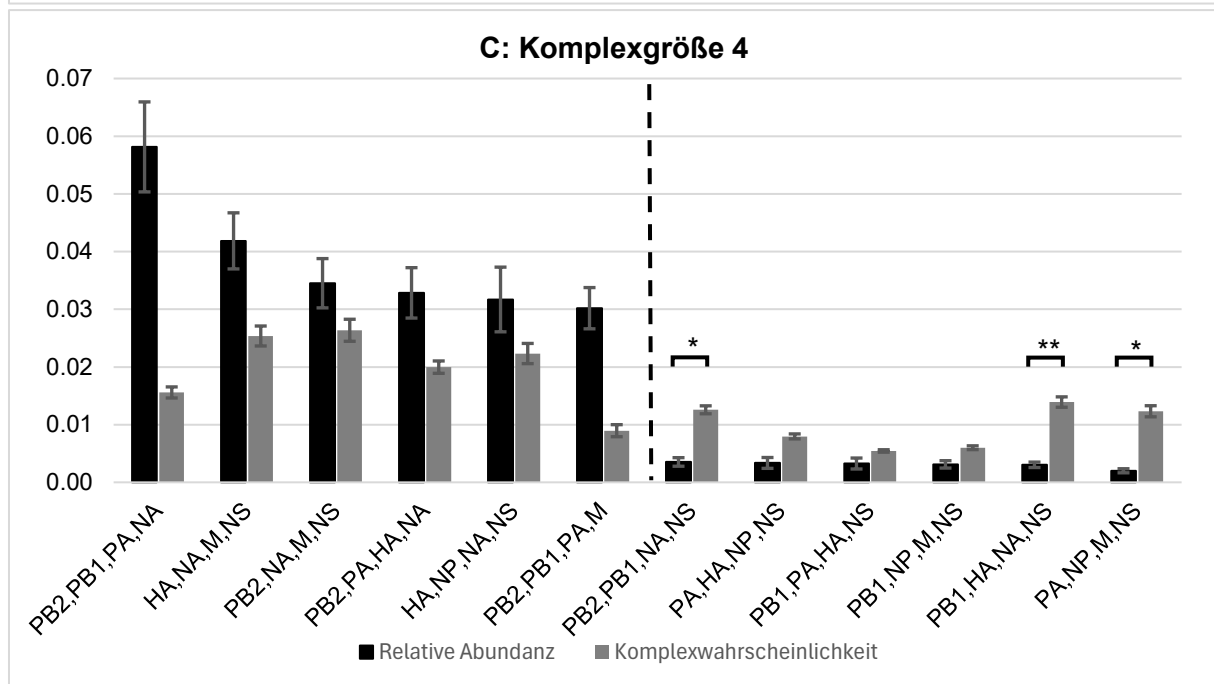
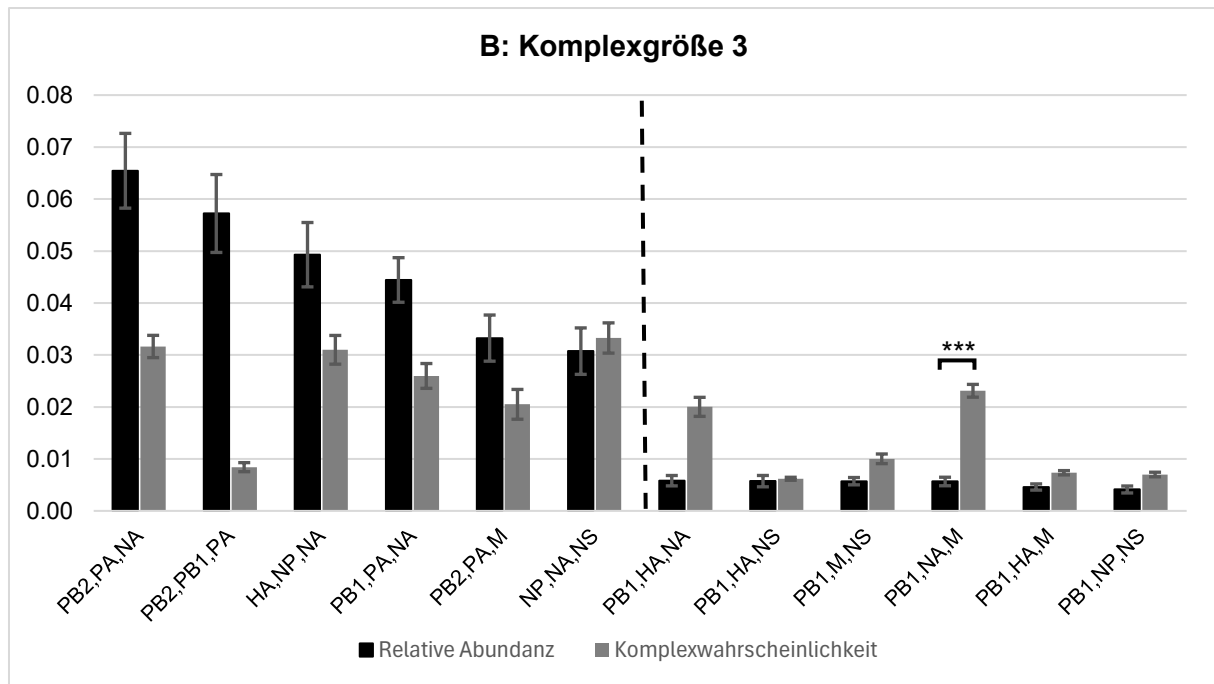
Die Abbildung (D) zeigt die relativen Abundanzen und Komplexwahrscheinlichkeiten der Fünferkomplexe. Die Ergebnisse der sechs abundantesten Komplexe zeigten, dass diese Komplexe häufiger vorlagen, als durch zufällige Assemblierung zu erwarten gewesen wäre. Die beiden abundantesten Komplexe waren HA,NP,NA,M,NS und PB2,PA,HA,NA,M. Die relativen Abundanzen betrugen 6,8 % und 6,0 %. Die Komplexwahrscheinlichkeiten betrugen jeweils circa drei Prozent und waren damit halb so groß. Die beiden Komplexe PB2,PA,HA,M,NS und PB2,HA,NP,NA,M hatten eine relative Abundanz von über fünf Prozent. Die Komplexwahrscheinlichkeiten der beiden Komplexe betrugen mit 1,7 % und 2,3 % weniger als die Hälfte. Von den sechs abundantesten Komplexen beinhalteten fünf Komplexe die Segmente HA und NA. Außerdem konnte die Kombination von M und NS in drei der sechs abundantesten Komplexe detektiert werden. Die relativen Abundanzen der seltensten Komplexe betrugen zwischen 0,2 % und 0,4 %, während die Wahrscheinlichkeiten mindestens doppelt so hoch waren. Bei den Komplexen PB2,PB1,NP,NA,NS; PB1,PA,HA,NA,NS; PB1,PA,HA,NA,M und PB1,HA,NA,M,NS waren die Unterschiede mit bis zu zehnfachen Wahrscheinlichkeiten signifikant.

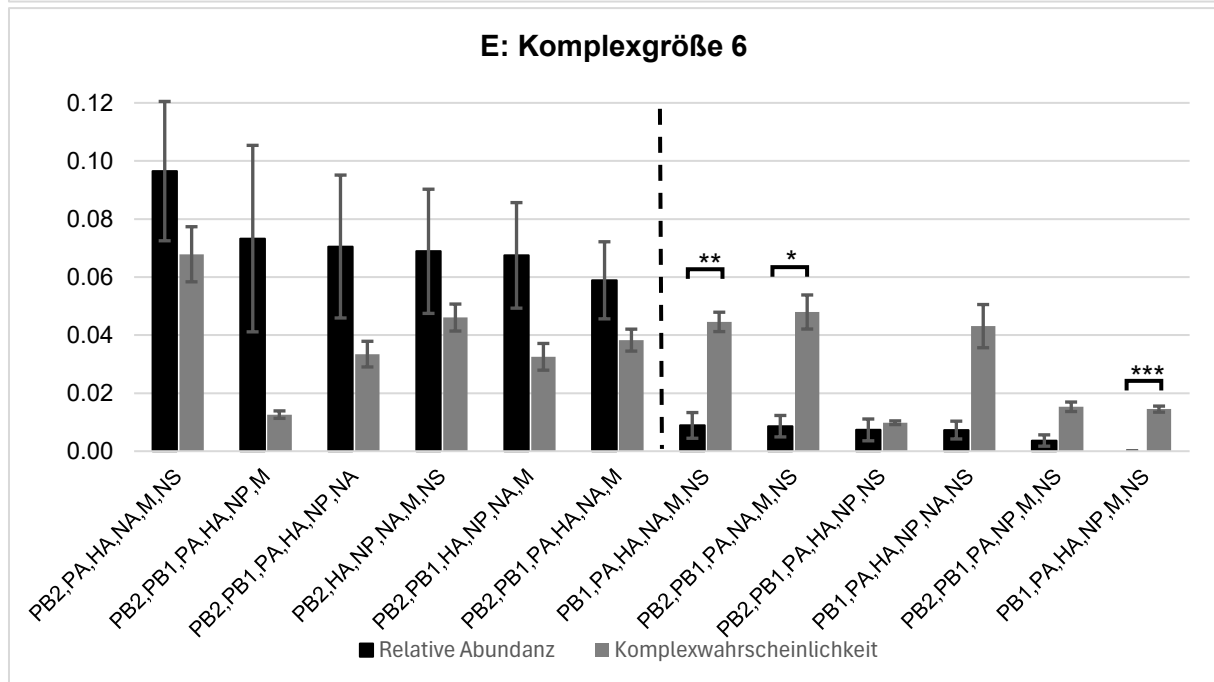
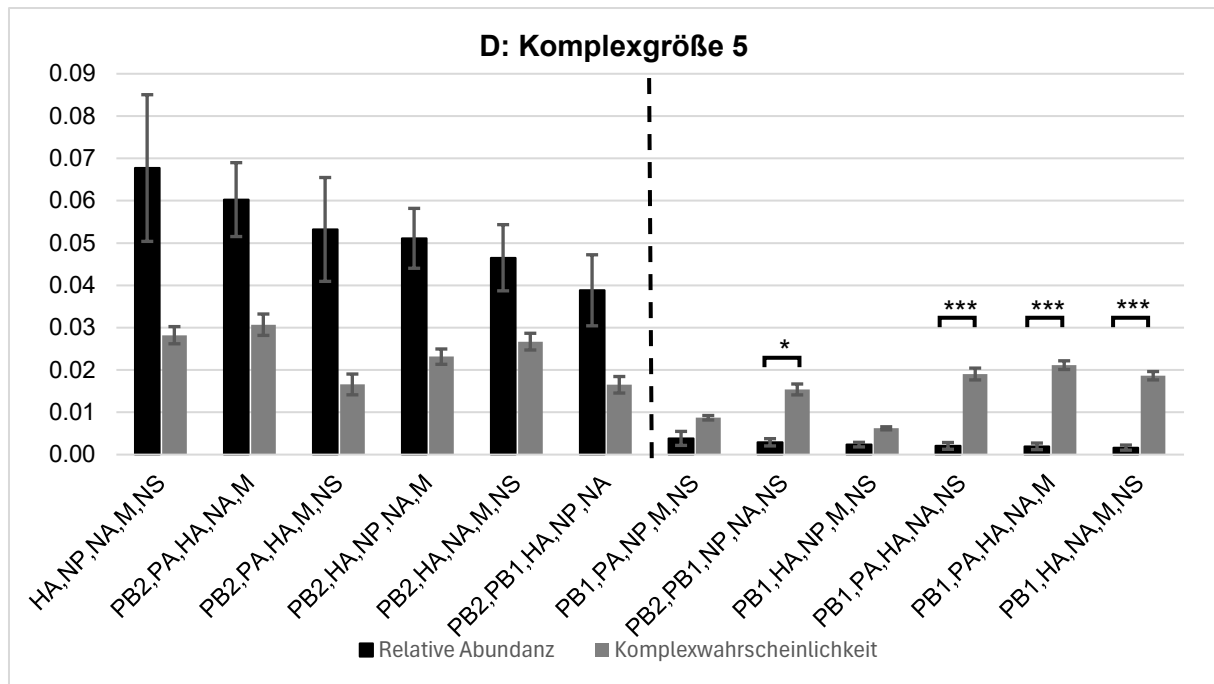
In Abbildung (E) sind die relativen Abundanzen und Komplexwahrscheinlichkeiten der Sechserkomplexe dargestellt. Am häufigsten wurde der Komplex PB2,PA,HA,NA,M,NS detektiert. Dieser hatte einen Anteil von knapp zehn Prozent aller Sechserkomplexe. Die anderen fünf häufigsten Komplexe hatten einen Anteil von jeweils sechs bis sieben Prozent. Alle Komplexe zeigten eine deutlich höhere Abundanz als Komplexwahrscheinlichkeit. Am größten war die Differenz bei dem Komplex PB2,PB1,PA,HA,NP,M. Die sechs seltensten Komplexe zeigten jeweils eine relative Abundanz von unter einem Prozent, während die Wahrscheinlichkeiten höher waren. Der Komplex PB1,PA,HA,NP,M,NS wurde gar nicht detektiert. Die größten Differenzen zwischen Abundanz und Wahrscheinlichkeit zeigten die Komplexe PB1,PA,HA,NA,M,NS; PB2,PB1,PA,NA,M,NS und PB1,PA,HA,NP,NA,NS, wobei

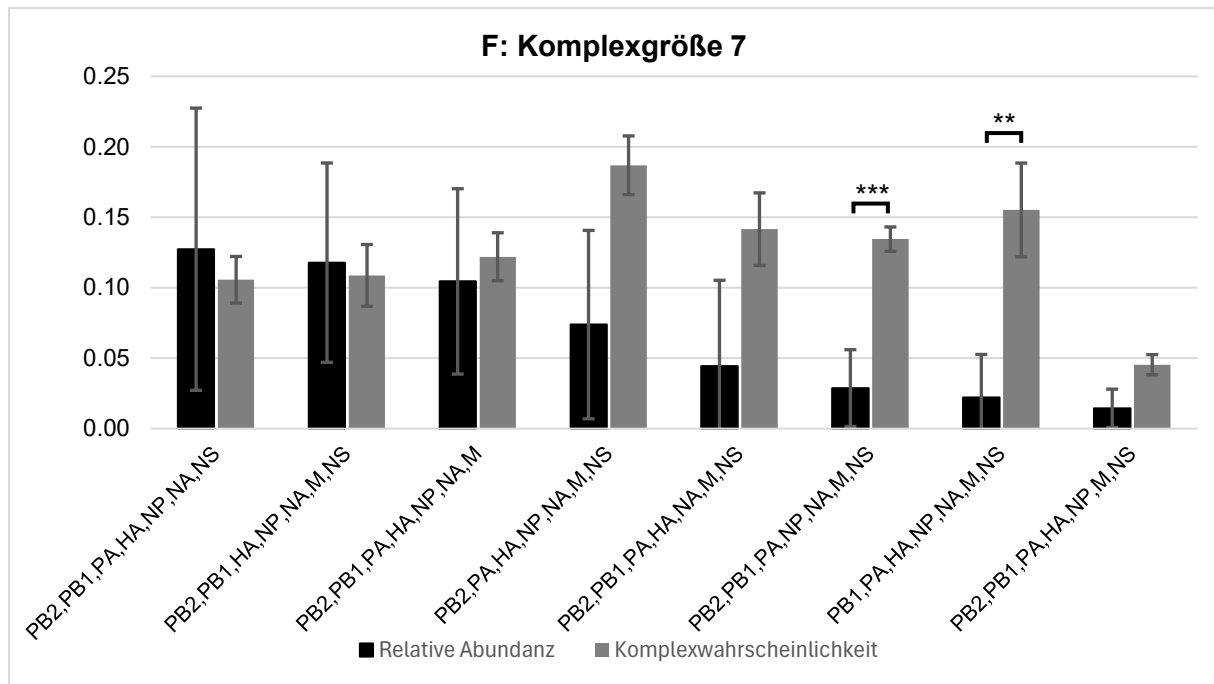
die Unterschiede zwischen den beiden erstgenannten Komplexen signifikant war. Insgesamt war auffällig, dass die drei Komplexe mit einer signifikant niedrigeren relativen Abundanz die Segmente M und NS beinhalten. Gleichzeitig enthielten aber auch zwei der abundantesten Komplexe beide Segmente.

Abbildung (F) zeigt die relativen Abundanzen und Komplexwahrscheinlichkeiten der Siebenerkomplexe. Die relative Abundanz war nur bei den beiden abundantesten Komplexen höher als die Wahrscheinlichkeit. Die Differenz war dabei gering. Bei den anderen sechs Komplexen war die relative Abundanz geringer als die Wahrscheinlichkeit. Signifikant waren die Differenzen bei den Komplexen PB2,PB1,PA,NP,NA,M,NS und PB1,PA,HA,NP,NA,M,NS. Damit konnte gezeigt werden, dass zwei der Siebenerkomplexe mit M und NS eine signifikant geringere Abundanz hatten, als durch die berechnete Wahrscheinlichkeit zu erwarten gewesen wäre. Gleichzeitig waren unter den drei abundantesten Komplexen zwei Komplexe vertreten, bei denen jeweils M oder NS fehlte. Auffällig waren zudem die insgesamt hohen Standardfehler.









**Abb. 21: Vergleich von relativer Abundanz und Komplexwahrscheinlichkeit pro Komplexgröße.**

Die Komplexwahrscheinlichkeiten der Komplexe wurden durch Multiplikation der relativen Häufigkeiten der Einzelsegmente berechnet. Alle Einzelsegmente und Komplexe wurden durch Spot-basierte Dekodierung in undichten Regionen detektiert. Jede Abbildung zeigt das arithmetische Mittel der relativen Abundanz (schwarz) und der Komplexwahrscheinlichkeit (grau) ( $n=15$ ) der fünf Replikate und Infektionszeitpunkte vier bis sechs. (A) zeigt die Ergebnisse der Komplexgröße zwei, (B) die der Größe drei, (C) die der Größe vier, (D) die der Größe fünf, (E) die der Größe sechs, (F) die der Größe sieben. Links der Trennlinie sind jeweils die sechs abundantesten Komplexe dargestellt. Rechts der Trennlinie werden die sechs wenig abundantesten Komplexe gezeigt. Die restlichen Komplexe befinden sich im Anhang (Abb. 27 - Abb. 31). Die Fehlerbalken zeigen die Standardfehler. Sternchen markieren die Signifikanz ( $\alpha = 0,05$ ), ermittelt per unabhängigem, zweiseitigem T-Test. Die berechneten p-Werte wurden nach der Methode von Bonferroni für multiples Testen korrigiert. (Bland und Altman 1995). Ein Sternchen markiert einen p-Wert kleiner gleich 0,05. Zwei Sternchen markieren einen p-Wert kleiner gleich 0,01 und drei Sternchen einen p-Wert kleiner gleich 0,001.

#### 4.2.3 Modellierung einer monomeren Assemblierung

Für die Bestimmung der wahrscheinlichsten monomeren Assemblierungswege der acht vRNP-Segmente wurden die relativen Häufigkeiten einzigartiger Komplexe zwischen vier bis sechs Stunden ermittelt. Alle Komplexe wurden durch Spot-basierte Dekodierung in undichten Regionen ermittelt. Anschließend wurden für alle 40.320 Permutationen die logarithmierten relativen Häufigkeiten der Zwischenkomplexe aufsummiert, um die Wahrscheinlichkeit jeder Permutation zu erhalten.

In **Tab. 3** sind die zehn wahrscheinlichsten Permutationen in absteigender Reihenfolge aufgelistet. Die Permutation NA,PA,PB2,PB1,HA,NP,M,NS war mit -16,57 am wahrscheinlichsten. Die Ergebnisse zeigten, dass die Segmente M und NS in sechs der zehn wahrscheinlichsten Permutationen direkt hintereinander assemblierten. Die vier wahrscheinlichsten Permutationen zeigten die Assemblierung der Segmente PB2, PB1 und PA direkt hintereinander oder unterbrochen durch das Segment NP. Insgesamt zeigten die Ergebnisse, dass die Wahrscheinlichkeiten der Permutationen annähernd normalverteilt waren. Die zehn Permutationen mit Wahrscheinlichkeiten kleiner als 17 stellten 0,02 % der gesamten Permutationen dar. Die Verteilung der Wahrscheinlichkeiten ist im Anhang abgebildet (**Abb. 32**).

**Tab. 3: Die zehn wahrscheinlichsten Permutationen der Assemblierung der Influenzasegmente und ihre Wahrscheinlichkeiten.** Die Permutationen sind nach absteigender Wahrscheinlichkeit sortiert. Die Wahrscheinlichkeiten wurden durch die Addition aller logarithmierten relativen Häufigkeiten der intermediären Komplexe berechnet. Die Wahrscheinlichkeiten wurden basierend auf den relativen Häufigkeiten der Infektionszeitpunkte vier bis sechs, detektiert mittels Spot-basierter Dekodierung in undichten Regionen, berechnet.

Permutation	Wahrscheinlichkeit
NA, PA, PB2, PB1, HA, NP, M, NS	-16.57
M, NS, NA, HA, PB2, NP, PB1, PA	-16.58
NA, PA, PB2, PB1, HA, M, NP, NS	-16.63
NS, M, NA, HA, PB2, NP, PB1, PA	-16.69
M, NS, NA, PB2, HA, NP, PB1, PA	-16.84
NA, PA, PB2, M, HA, PB1, NP, NS	-16.85
NA, PA, PB2, HA, M, PB1, NP, NS	-16.88
NS, M, NA, PB2, HA, NP, PB1, PA	-16.95
NA, NP, HA, PB2, PB1, M, NS, PA	-16.99
NA, PA, PB2, M, HA, NS, NP, PB1	-17.00

## 5. Diskussion

### 5.1 Pipeline Qualität und Verbesserungen

Im Rahmen dieser Arbeit wurde eine automatisierte Analysepipeline zur Auswertung eines In-situ Sequenzierungsdatensatzes entwickelt und evaluiert. Die Ergebnisse belegen, dass die Pipeline grundsätzlich geeignet ist Spots konsistent zu detektieren und zu dekodieren. Vergleiche mit manuellen Annotationen haben jedoch gezeigt, dass insbesondere die Dekodierung noch weiter optimiert werden sollte.

Der Vergleich von manuell festgelegten und modellbasierten Schwellenwerten zur Maskierung von dichten und undichten Regionen zeigte, dass die Bestimmung eines Schwellenwerts durch ein Regressionsmodell zuverlässig funktionierte (**Abb. 9**). Der Schwellenwert wurde durch ein Perzentil der Intensitätswerte festgelegt. In der Analyse wurden die mittleren, manuell festgelegten und modellbasierten Perzentile aller FOVs eines Infektionszeitpunkts verglichen. Im Durchschnitt betrug die Abweichung zwischen den manuellen und modellbasierten Perzentilen 1,7 %. Vor allem zu frühen und mittleren Infektionszeiten waren die Abweichungen sehr gering. Zu späten Infektionszeiten waren die Abweichungen größer. Das lag daran, dass die Abweichungen zwischen den einzelnen FOVs innerhalb eines Infektionszeitpunkts zunahmten. Zu sehen ist die erhöhte Varianz an den größeren Standardabweichungen. Da die Intensitätswerte zu späteren Infektionszeitpunkten allgemein höher waren, wirkte sich eine größere Abweichung des Perzentils weniger stark aus als zu frühen Infektionszeiten, wo die Intensitäten insgesamt niedriger waren. Dieser Zusammenhang konnte während der manuellen Festlegung der Schwellenwerte und bei der Überprüfung der modellbasierten Schwellenwerte festgestellt werden. Bei den Aufnahmen von null bis drei Stunden nach Infektion wurden die Perzentile daher einheitlich auf 100 % gesetzt, sodass keine dichten Regionen maskiert wurden.

Der Vergleich der Spotanzahlen von manuell annotierten und durch Spot-basierte Dekodierung (SBD) detektierten Spots zeigte, dass die SBD 50 % mehr Spots detektierte (**Abb. 10**). Während die Gesamtzahl der manuell annotierten Spots 927 betrug, detektierte die SBD 1530 Spots. Davon wurden 667 Spots sowohl durch die manuelle Annotation als auch durch die SBD detektiert. In Replik 0 war die Gesamtzahl der manuell annotierten Spots höher als die der SBD. In den Replik 1 und 2 hingegen, annotierte die SBD mehr als dreimal so viele Spots. Die hohen Spotanzahlen der SBD in Replik 1 und 2 können durch die hohe Anzahl von Spots, die als „invalide“ annotiert wurden, erklärt werden. Es wurden 820, der insgesamt 1530 Spots, als „invalide“ annotiert. Die deutlich höhere Anzahl von als „invalide“ annotierten Spots in Replik 1 und 2 durch die Pipeline, kann durch die unterschiedliche interne Aufbereitung der Bilder durch die Software des Mikroskops erklärt werden. Ein Update zwischen Replik 0 und den restlichen Replik 1 und 2 bewirkte, dass die Spots in den Aufnahmen weniger intensiv und diffuser waren (**Abb. 4**). Dies erschwerte die Dekodierung der Pipeline.

Für eine detailliertere Bewertung der SBD wurden die Barcodes der übereinstimmend detektierten Spots aus manueller Annotation und der SBD verglichen. Der Vergleich der Barcodes zeigte, dass 67 % der Spots gleich dekodiert wurden (**Abb. 11**). Zusätzlich wurden alle Spots, bei denen die Dekodierung der manuellen Annotation und der SBD nicht übereinstimmten analysiert, um Fehlkodierungen auf Ebene einzelner Basen zu bewerten



(**Abb. 12**). Im Vergleich zu den anderen Basen war die Anzahl der Fehlkodierungen bei keiner Base auffällig groß. Durch die Betrachtung der Barcodes (**Tab. 2**) wäre zu erwarten gewesen, dass in der ersten Sequenzierunde die Fehldekodierungen von Thymin am höchsten sind, in der zweiten Runde von Guanin und in der dritten Runde von Cytosin. Für die ersten beiden Runden stimmten die Ergebnisse mit den Erwartungen überein. Außerdem zeigte sich keine fehlerhaft detektierte Guanin Base in der dritten Runde. Dieses Ergebnis war zu erwarten, da keiner der Barcodes Guanin an dritter Position enthielt. Die Ergebnisse der basenspezifischen Barcodeanalyse konnte somit zeigen, dass keine Base durch die SBD systematisch falsch detektiert wurde.

Da die konsistente Dekodierung von Spots eine wichtige Voraussetzung für die korrekte Dekodierung ist, wurde die Konsistenz der Spot-basierten Dekodierung in dichten und undichten Regionen von überlappenden FOV-Bereichen analysiert (**Abb. 13**). Die Spots aus diesen Bereichen wurden zweimal aufgenommen und eigneten sich daher gut zur Überprüfung der konsistenten Detektion durch die SBD (**Abb. 3**). Für die Bewertung wurden Spots aus benachbarten FOVs mit übereinstimmenden Positionen gepaart und die Barcodedistanz berechnet. In den dichten Bereichen wurden 80.868 Spotpaare detektiert, die zu unter 50 % gleich dekodiert wurden. In den undichten Regionen wurden 51.528 Spotpaare detektiert, die zu 84 % konsistent dekodiert wurden. Die wesentlich höhere Konsistenz in den undichten Bereichen zeigte, dass die SBD besser für undichte Regionen geeignet ist und für dichte Regionen unzureichende Ergebnisse lieferte. Auch wenn eine konsistente Dekodierung nicht automatisch auch eine korrekte Dekodierung bedeutet, deutete die hohe Konsistenz in undichten Regionen auf eine zuverlässige Detektion und Dekodierung hin. Die geringe Konsistenz in undichten Regionen dagegen ist ein deutlicher Hinweis auf eine fehlerhafte Detektion und Dekodierung. Zusätzlich wurde beobachtet, dass die Registrierung der überlappenden Bereiche aufgrund der kleinen Fläche teilweise unzureichend war. Das hatte zur Folge, dass die Positionen der Spots aus einigen Überlappungen nicht richtig angeglichen werden konnten. Abschließend zeigten die Ergebnisse der SBD, dass für die Dekodierung in dichten Regionen eine alternative Strategie wie die Pixel-basierte Dekodierung (PBD) nötig war.

Daher wurde das PBD mit in die Analysepipeline integriert. Der Vergleich der relativen Segmentanzahlen, die in undichten Regionen mittels manueller Dekodierung, SBD und PBD ermittelt wurden, zeigte vergleichbare Ergebnisse der beiden Dekodierungsmethoden (**Abb. 14**). Insgesamt stimmten jedoch die Ergebnisse der PBD etwas besser mit der manuellen Annotation überein. Diese Ergebnisse deuteten auf eine genauere Dekodierung der PBD in undichten Regionen hin. In einer detaillierteren Analyse wurden die detektierten Spots der PBD aus undichten Regionen mit der manuellen Annotation verglichen. Dabei wurde wie bei dem

Vergleich der manuellen Annotation mit der SBD vorgegangen (siehe 4.1.2.1). Die Anzahlen von detektierten Spots pro FOV waren insgesamt niedriger als bei der manuellen Annotation (**Abb. 15**). Im Vergleich zur SBD betrug die Anzahl weniger als die Hälfte (**Abb. 10**). Somit waren auch die Anzahlen von übereinstimmend detektierten Spots in der PBD niedriger. Die Betrachtung der Barcodedistanzen zwischen gepaarten Spots aus der manuellen Annotation und PBD ergab, dass 35 % der Spots gleich dekodiert wurden (**Abb. 16**). Im Vergleich zur SBD, wo 67 % der Spots übereinstimmend detektiert wurden, war die Korrektheit gering (**Abb. 11**). Aufgrund der höheren Anzahl von nicht-übereinstimmend detektierten Spots waren auch die Anzahlen fehlerhaft detektierter Basen höher (**Abb. 17**). Es zeigten sich Fehldetektionen, die durch den Aufbau der Barcodes nicht erklärt werden konnten (**Tab. 2**). Die Base Adenin wurde am häufigsten fehlerhaft dekodiert, obwohl die Base am seltensten in den Barcodes vorkam. Das deutete auf eine systematische Fehldetektion von Adenin durch die PBD hin. Außerdem wurde Thymin am häufigsten in der dritten Sequenzierrunde falsch detektiert, obwohl die meisten Barcodes Thymin an erster Stelle enthielten. Zusammenfassend lässt sich sagen, dass die PBD in undichten Regionen, bezogen auf Quantität und Qualität der Dekodierung, deutlich schlechtere Ergebnisse als die SBD erzielte.

Die Dekodierungsqualität der PBD in dichten Regionen konnte nicht anhand der manuellen Referenz bewertet werden, da eine manuelle Unterscheidung einzelner Spots in diesen Regionen nicht möglich war. Daher wurden die relativen Segmentanteile der SBD aus undichten Regionen und PBD aus dichten Regionen verglichen (**Abb. 18**). Die Segmentanteile waren größtenteils vergleichbar. Lediglich bei NA lag der Anteil in der SBD deutlich höher als in der PBD und bei den Segmenten HA und NP waren die Anteile der PBD höher. Dabei ist jedoch zu beachten, dass die Expression der RNAs und Formation der Segmente im Infektionsverlauf variiert (Ahmad et al. 2025). Da dichte Regionen typischerweise erst zu späteren Infektionszeiten auftraten, waren Unterschiede in den Segmentanzahlen nicht zwingend auf Unterschiede in der Dekodierungsqualität zurückzuführen.

Eine detailliertere Analyse der PBD, in der die konsistente Dekodierung in überlappenden FOV-Bereichen, analysiert wurde, zeigte, dass die PBD insgesamt inkonsistent war (**Abb. 19**). Während der Analyse wurde wie bei der Bewertung der SBD vorgegangen (siehe 4.1.2.4). In dichten Regionen lag der Anteil an übereinstimmend dekodierten Spots bei 18 % (A). In undichten Regionen betrug er 58 % (B). Damit konnte gezeigt werden, dass auch die PBD keine konsistente Dekodierung in dichten Regionen ermöglichte. Insgesamt wurde gezeigt, dass die PBD in Bezug auf Konsistenz und Korrektheit, sowohl in undichten als auch dichten Regionen, schlechtere Dekodierungsergebnisse als die SBD erzielte.

Abschließend lässt sich festhalten, dass keine der beiden Dekodierungsmethoden geeignet war alle Regionen zu dekodieren. Sowohl die SBD als auch die PBD lieferten in dichten

Regionen inkonsistente Ergebnisse. In den undichten Regionen erzielte die SBD vergleichsweise gute Ergebnisse, während die PBD dort schlechter abschnitt. Außerdem ist zu beachten, dass auch die manuelle Annotation, welche als Referenz diente, nicht frei von fehlerhaften oder inkonsistenten Ergebnissen ist. Besonders bei überlagernden oder schwachen Spots war eine konsistente Annotation schwer. Positiv ist jedoch, dass alle manuellen Annotationen von einer Person durchgeführt wurden, wodurch eine konsistentere Annotation ermöglicht wurde. Für die weitere Auswertung des Datensatzes sollten daher nur Ergebnisse der SBD aus undichten Regionen beachtet werden, da die Zuverlässigkeit dort am höchsten war.

## 5.2 Möglicher Mechanismus der Genomassemblierung

Die Analyse der möglichen Assemblierungswege des Influenzagenoms zeigt, dass die Genomassemblierung nicht rein zufällig erfolgt. Die Bildung eines vollständigen Genomkomplexes mit allen acht Segmenten scheint über die Bildung von Subkomplexen zu verlaufen, wobei bestimmte Komplexe bevorzugt gebildet werden. Diese Ergebnisse konnten übereinstimmend durch Betrachtung von Komplexanzahlen, Vergleiche zwischen beobachteten Abundanzen und berechneten Komplexwahrscheinlichkeiten und durch die Modellierung einer monomeren Assemblierung gezeigt werden.

Die relativen Anzahlen der Einzelsegmente und Komplexe zeigten, dass Komplexe, die ein oder mehrere der Polymerasesegmente PB2, PB1 oder PA enthielten, am häufigsten detektiert wurden (**Abb. 20**). PB2, PB1 und PA waren die abundantesten Einzelsegmente (A) und PB2,PB1 und PB2,PA die zwei abundantesten Zweierkomplexe (B). Der abundanteste Dreierkomplex war PB2,PB1,PA (C). Außerdem enthielten die elf abundantesten Dreierkomplexe zwei der drei Polymerasesegmente (**Abb. 28**). Von den acht abundantesten Vierer- bis Siebenerkomplexen (D-G) enthielten jeweils alle die Segmente PB2, PB1 und PA. Diese Ergebnisse wiesen auf eine bevorzugte, frühe Assemblierung der drei Segmente hin. Jedoch sollte beachtet werden, dass die beschriebenen Anzahlen das arithmetische Mittel der fünf Replikate aus drei verschiedenen Infektionszeiten darstellen ( $n = 15$ ). Die zum Teil hohe Varianz und ungleiche Verteilung der Einzelwerte zwischen den Replikaten und Zeitpunkten wies auf dynamische und variierende Assemblierungen hin. Die wiederholte Co-Lokalisation von den Polymerasesegmenten über verschiedene Komplexgrößen war ein konsistentes Muster und lässt sich gut mit der bekannten Funktionen der Segmente in Einklang bringen. Da PB2, PB1 und PA die Untereinheiten der viralen Polymerase darstellen, ist die gemeinsame Assemblierung biologisch sinnvoll (Wandzik et al. 2021). Aus früheren Studien war bekannt, dass Mutationen in Verpackungssignalen der Polymerase sowohl die eigene Verpackung als auch die Verpackung anderer Segmente negativ beeinflussten (Gao et al. 2012). Besonders

die Mutation von PB2 führte häufig zu einer reduzierten Verpackung von PB1, PA, NP, M und NS (Marsh et al. 2008). Diese Ergebnisse wiesen auf eine zentrale Rolle von PB2 während der Assemblierung hin. Die gegenseitige Abhängigkeit der Polymerasesegmente PB2, PB1 und PA sprach für eine gemeinsame Assemblierung zu frühen Infektionszeitpunkten.

Komplexe, die zwei oder alle drei Polymerasesegmente PB2, PB1 und PA enthielten, wurden öfter detektiert, als es bei rein zufälliger Assemblierung zu erwarten gewesen wäre (**Abb. 21**). Außerdem wich die relative Abundanz von Komplexen mit M und NS häufig signifikant von erwarteten Abundanzen ab. Dies zeigte sich im Vergleich der beobachteten relativen Abundanzen mit den theoretisch berechneten Komplexwahrscheinlichkeiten, welche durch Multiplikation der relativen Abundanzen der Einzelsegmente berechnet wurden. Zudem konnten während der Analyse weitere Komplexe identifiziert werden, die eher selten auftraten und wahrscheinlich nicht Teil gängiger Assemblierungswege sind. Bei den Zweierkomplexen zeigten insbesondere PB2,PA und PB1,PA die größten Abweichungen zwischen beobachteter Abundanz und berechneter Wahrscheinlichkeit (A), wobei die beobachtete Abundanz deutlich höher war. Umgekehrt wiesen viele Zweierkomplexe aus einem Polymerasesegment und einem Nicht-Polymerasesegment niedrigere Abundanzen auf als erwartet. Bei den Dreierkomplexen lag die größte Differenz zwischen relativer Abundanz und Wahrscheinlichkeit bei dem Komplex PB2,PB1,PA (B), wobei die relative Abundanz deutlich höher war. Auch bei den Vierer- und Sechserkomplexen zeigten die Komplexe, die alle drei Polymerasesegmente enthielten, die höchsten Differenzen zwischen Abundanz und Wahrscheinlichkeit (C, E). Bei den Siebenerkomplexen gehörte der Komplex, dem PB2 fehlte, zu den am seltensten detektierten (F). Zudem wies er die größte negative Abweichung von der berechneten Wahrscheinlichkeit auf. Weiterhin wurde der Komplex M,NS identifiziert, dessen relative Abundanz die berechnete Wahrscheinlichkeit deutlich übertraf (A). Zudem wurde beobachtet, dass viele der Vierer- bis Siebenerkomplexe mit den größten Differenzen zwischen Abundanz und Wahrscheinlichkeit, die Segmente M und NS enthielten (D-G). Die Segmente traten dabei sowohl in Komplexen auf, die deutlich seltener detektiert wurden als erwartet, als auch bei Komplexen, die deutlich abundanter waren. Das häufige Auftreten von Komplexen mit Polymerasesegmenten deutete auf eine bevorzugte Assemblierung der Segmente und auf eine zentrale Rolle innerhalb der Assemblierung hin. Diese Ergebnisse waren somit mit den Ergebnissen der Komplexanzahlen übereinstimmend (**Abb. 20**). Die hohe Abundanz des Zweierkomplex M,NS und das häufige Auftreten von Komplexen mit M- und NS-Segmenten ließ ebenfalls auf eine frühe und zentrale Assemblierung schließen. Die gleichzeitig signifikant niedrigeren Abundanzen der Vierer- bis Siebenerkomplexe mit M und NS, deuteten darauf hin, dass viele der früh assemblierten M,NS Komplexe nicht weiter assemblierten. Gestützt wird diese These durch die Tatsache, dass die Sequenzen der beiden Segmente kurz sind und daher weniger RNA-RNA Interaktionen zwischen den Segmenten ermöglichen. Außerdem

konnte bereits gezeigt werden, dass bereits die Mutation weniger Nukleotide im Segment M die Viruspopulation stark senkte (Hutchinson et al. 2008). Die hohe Abundanz des Zweierkomplexes M,NS konnte bereits gezeigt werden (Hepp et al. 2025).

Die Modellierung monomerer Assemblierungswege ergab, dass verschiedene Permutationen der Segmentassemblierung nahezu gleich wahrscheinlich waren (**Tab. 3**). Die Berechnung erfolgte durch die Addition der relativen Abundanzen der intermediären Komplexe. In den Permutationen waren bestimmte Muster von gemeinsam auftretenden Segmenten zu erkennen. Insbesondere die Polymerasesegmente PB2, PB1 und PA sowie die Segmente M und NS traten häufig zusammen auf. Damit bestätigte diese Analyse die Ergebnisse der Komplexanzahlen und berechneten Komplexwahrscheinlichkeiten. Das gemeinsame Auftreten dieser Komplexe konnte in den beiden Analysen bereits gezeigt werden. In der früheren Studie von Hepp et al. wurde gezeigt, dass in Komplexen mit nur einem fehlenden Segment, am häufigsten kein NS vorhanden war und daher wahrscheinlich als letztes assemblierte. Die wahrscheinlichste Permutation zeigte ebenfalls, dass NS als letztes assemblierte. Zudem zeigten drei weitere der zehn wahrscheinlichsten Permutationen dieses Ergebnis. Jedoch ist zu beachten, dass für die Modellierung aus Gründen der Simplifikation ein monomerer Assemblierungsweg angenommen wurde. Dies bildet die Realität nur bedingt ab, da bereits gezeigt werden konnte, dass die Assemblierung des Genoms über die Bildung von Subkomplexen abläuft (Bouvier und Palese 2008). Dennoch lieferte die Modellierung Hinweise auf bevorzugte Interaktionen, die wahrscheinlich auch in der Assemblierung von Subkomplexen eine Rolle spielen.

Die vorliegenden Ergebnisse der beobachteten Abundanzen und berechneten Komplexwahrscheinlichkeiten, sowie der Modellierung, zeigen deutlich, dass die Verpackung des Genoms weder rein zufällig noch strikt linear verläuft. Die Assemblierung der Segmente scheint entlang mehrerer gleichzeitig existierender Permutationen zu verlaufen. Durch die bevorzugte Interaktion einzelner Segmente über spezifische RNA-RNA Interaktionen werden bestimmte Subkomplexe bevorzugt gebildet. Die Verpackung des Influenzagenoms zur Bildung vollständiger viraler Partikel ist dynamisch und erfolgt über selektive Interaktionen.

## 5.3 Ausblick

Die in dieser Arbeit entwickelte Analysepipeline stellt eine gute Grundlage für die automatisierte Auswertung von In-situ Sequenzierungen dar. Dennoch besteht weiter Potenzial bei der Optimierung der Dekodierung, insbesondere in dichten Regionen. Zudem wurden umfassende Validierungen durchgeführt, die auch für die Bewertung weiterer In-situ Daten geeignet sind.

Weiterhin sollte die Analysepipeline hinsichtlich ihrer Übertragbarkeit auf andere Datensätze getestet werden.

Im Hinblick auf die Genomassemblierung lieferte diese Arbeit Hinweise auf nicht-zufällige, dynamische Assemblierungswege. In künftigen Analysen können diese Erkenntnisse vertieft werden, indem die Assemblierung detaillierter über weitere Infektionszeitpunkte analysiert wird. Mithilfe komplexerer Modellierungen, die auch die Assemblierungen über Subkomplexe betrachten, könnten weitere Erkenntnisse über die selektiven Verpackungsmechanismen gewonnen werden. Zusätzlich würde die Analyse weiterer Virusstämme und Mutanten dazu beitragen, konservierte Mechanismen zu identifizieren.

Langfristig können die Entwicklung automatisierter Analyseverfahren und das Verständnis der Genomassemblierung dazu beitragen, die Entwicklung von antiviralen Medikamenten oder Impfstoffen voranzutreiben.

## 6. Referenzen

- Ahmad, Shazeb; Li, Jianhui; Schaust, Jo  l; Gribling-Burrer, Anne-Sophie; Geiger, Nina; Fischer, Sabine C. et al. (2025): Visualizing the transcription and replication of influenza A viral RNAs in cells by multiple direct RNA padlock probing and in situ sequencing (mudRapp-seq). In: *Nucleic acids research* 53 (11). DOI: 10.1093/nar/gkaf461.
- Amorim, Maria Joao; Bruce, Emily A.; Read, Eliot K. C.; Foeglein, Agnes; Mahen, Robert; Stuart, Amanda D.; Digard, Paul (2011): A Rab11- and microtubule-dependent mechanism for cytoplasmic transport of influenza A virus viral RNA. In: *Journal of virology* 85 (9), S. 4143–4156. DOI: 10.1128/jvi.02606-10.
- Axelrod et al. (2018): Starfish: Open Source Image Based Transcriptomics and Proteomics Tools. Online verf  gbar unter <http://github.com/spacetx/starfish>, zuletzt gepr  ft am 19.06.25.
- Bai, Xiangzhi; Zhou, Fugen; Xue, Bindang (2012): Toggle and top-hat based morphological contrast operators. In: *Computers & Electrical Engineering* 38 (5), S. 1196–1204. DOI: 10.1016/j.compeleceng.2011.08.010.
- Bland, J. M.; Altman, D. G. (1995): Multiple significance tests: the Bonferroni method. In: *BMJ (Clinical research ed.)* 310 (6973), S. 170. DOI: 10.1136/bmj.310.6973.170.
- Bouvier, Nicole M.; Palese, Peter (2008): The biology of influenza viruses. In: *Vaccine* 26 Suppl 4 (Suppl 4), D49-53. DOI: 10.1016/j.vaccine.2008.07.039.
- D'Haeyer, Johannes P.F. (1989): Gaussian filtering of images: A regularization approach. In: *Signal Processing* 18 (2), S. 169–181. DOI: 10.1016/0165-1684(89)90048-0.
- Eisfeld, Amie J.; Neumann, Gabriele; Kawaoka, Yoshihiro (2015): At the centre: influenza A virus ribonucleoproteins. In: *Nature reviews. Microbiology* 13 (1), S. 28–41. DOI: 10.1038/nrmicro3367.
- Elsdoerfer, Michael (2011): glob2. Online verf  gbar unter <http://github.com/miracle2k/python-glob2/>, zuletzt gepr  ft am 05.09.2024.
- Gamblin, Steven J.; Skehel, John J. (2010): Influenza hemagglutinin and neuraminidase membrane glycoproteins. In: *The Journal of biological chemistry* 285 (37), S. 28403–28409. DOI: 10.1074/jbc.R110.129809.
- Gao, Qinshan; Chou, Yi-Ying; Do  anay, Sultan; Vafabakhsh, Reza; Ha, Taekjip; Palese, Peter (2012): The influenza A virus PB2, PA, NP, and M segments play a pivotal role during genome packaging. In: *Journal of virology* 86 (13), S. 7043–7051. DOI: 10.1128/JVI.00662-12.

- Hamilton, Brian S.; Whittaker, Gary R.; Daniel, Susan (2012): Influenza virus-mediated membrane fusion: determinants of hemagglutinin fusogenic activity and experimental approaches for assessing virus fusion. In: *Viruses* 4 (7), S. 1144–1168. DOI: 10.3390/v4071144.
- Harris, Charles R.; Millman, K. Jarrod; van der Walt, Stéfan J.; Gommers, Ralf; Virtanen, Pauli; Cournapeau, David et al. (2020): Array programming with NumPy. In: *Nature* 585 (7825), S. 357–362. DOI: 10.1038/s41586-020-2649-2.
- Harris C. R. et al. (2020): Array programming with NumPy (585). In: *Nature*, 2020, S. 357–362.
- Hepp, Christof; Zhao, Qing; Robb, Nicole; Fodor, Ervin; Kapanidis, Achillefs N. (2025): High-throughput Single-Virion DNA-PAINT Reveals Structural Diversity, Cooperativity and Flexibility during Selective Packaging in Influenza.
- Hunter, J. D. (2007): Matplotlib: A 2D graphics environment. In: *Computing in Science & Engineering*, 9(3), 2007, S. 90–95.
- Hutchinson, Edward C.; Curran, Martin D.; Read, Eliot K.; Gog, Julia R.; Digard, Paul (2008): Mutational analysis of cis-acting RNA signals in segment 7 of influenza A virus. In: *Journal of virology* 82 (23), S. 11869–11879. DOI: 10.1128/JVI.01634-08.
- Joél Schaust (2024): Integrated Pipeline for Whole-Cell and Nuclear Segmentation in In-situ Sequencing Data. Masterarbeit. Julius-Maximilians-Universität, Würzburg. Center for Computational and Theoretical Biology, zuletzt geprüft am 13.07.2025.
- Ke, Rongqin; Mignardi, Marco; Pacureanu, Alexandra; Svedlund, Jessica; Botling, Johan; Wählby, Carolina; Nilsson, Mats (2013): In situ sequencing for RNA analysis in preserved tissue and cells. In: *Nature methods* 10 (9), S. 857–860. DOI: 10.1038/nmeth.2563.
- Lohman, Gregory J. S.; Zhang, Yinhua; Zhelkovsky, Alexander M.; Cantor, Eric J.; Evans, Thomas C. (2014): Efficient DNA ligation in DNA-RNA hybrid helices by Chlorella virus DNA ligase. In: *Nucleic acids research* 42 (3), S. 1831–1844. DOI: 10.1093/nar/gkt1032.
- Marsh, Glenn A.; Rabadán, Raúl; Levine, Arnold J.; Palese, Peter (2008): Highly conserved regions of influenza a virus polymerase gene segments are critical for efficient viral RNA packaging. In: *Journal of virology* 82 (5), S. 2295–2304. DOI: 10.1128/jvi.02267-07.
- Max Planck Institute Magdeburg (Hg.) (30.06.25): Imaging of Influenza Virus Replication to Optimize Cell Culture-based Vaccine Production. Online verfügbar unter <https://www.mpi->



magdeburg.mpg.de/3022112/Imaging\_of\_influenza\_virus\_replication\_to\_optimize\_cell-culture\_based\_vaccine\_production, zuletzt aktualisiert am 30.06.25, zuletzt geprüft am 30.06.25.

McKinney, W. (2010): Data structures for statistical computing in python. In Proceedings of the 9th Python in Science Conference., 2010, S. 51–56.

Noda, Takeshi (2021): Selective Genome Packaging Mechanisms of Influenza A Viruses. In: *Cold Spring Harbor perspectives in medicine* 11 (7). DOI: 10.1101/cshperspect.a038497.

Noda, Takeshi; Kawaoka, Yoshihiro (2010): Structure of influenza virus ribonucleoprotein complexes and their packaging into virions. In: *Reviews in medical virology* 20 (6), S. 380–391. DOI: 10.1002/rmv.666.

Noda, Takeshi; Sagara, Hiroshi; Yen, Albert; Takada, Ayato; Kida, Hiroshi; Cheng, R. Holland; Kawaoka, Yoshihiro (2006): Architecture of ribonucleoprotein complexes in influenza A virus particles. In: *Nature* 439 (7075), S. 490–492. DOI: 10.1038/nature04378.

O'Neill, R. E.; Jaskunas, R.; Blobel, G.; Palese, P.; Moroianu, J. (1995): Nuclear import of influenza virus RNA can be mediated by viral nucleoprotein and transport factors required for protein import. In: *The Journal of biological chemistry* 270 (39), S. 22701–22704. DOI: 10.1074/jbc.270.39.22701.

Pedregosa, Fabian; Varoquaux, Gaël; Gramfort, Alexandre; Michel, Vincent; Thirion, Bertrand; Grisel, Olivier et al. (2012): Scikit-learn: Machine Learning in Python. DOI: 10.48550/arXiv.1201.0490.

Pflug, Alexander; Guilligay, Delphine; Reich, Stefan; Cusack, Stephen (2014): Structure of influenza A polymerase bound to the viral RNA promoter. In: *Nature* 516 (7531), S. 355–360. DOI: 10.1038/nature14008.

Sofroniew, Nicholas; Lambert, Talley; Bokota, Grzegorz; Nunez-Iglesias, Juan; Sobolewski, Peter; Sweet, Andrew et al. (2025): napari: a multi-dimensional image viewer for Python: Zenodo.

Te Velthuis, Aartjan J. W.; Fodor, Ervin (2016): Influenza virus RNA polymerase: insights into the mechanisms of viral RNA synthesis. In: *Nature reviews. Microbiology* 14 (8), S. 479–493. DOI: 10.1038/nrmicro.2016.87.

Van der Walt et. al. (2014): scikit-image: image processing in Python (2). In: *PeerJ*, 2014, S. 453.

van Rossum, G. (2020a): The Python Library Reference. Version 3.8.2: Python Software Foundation.

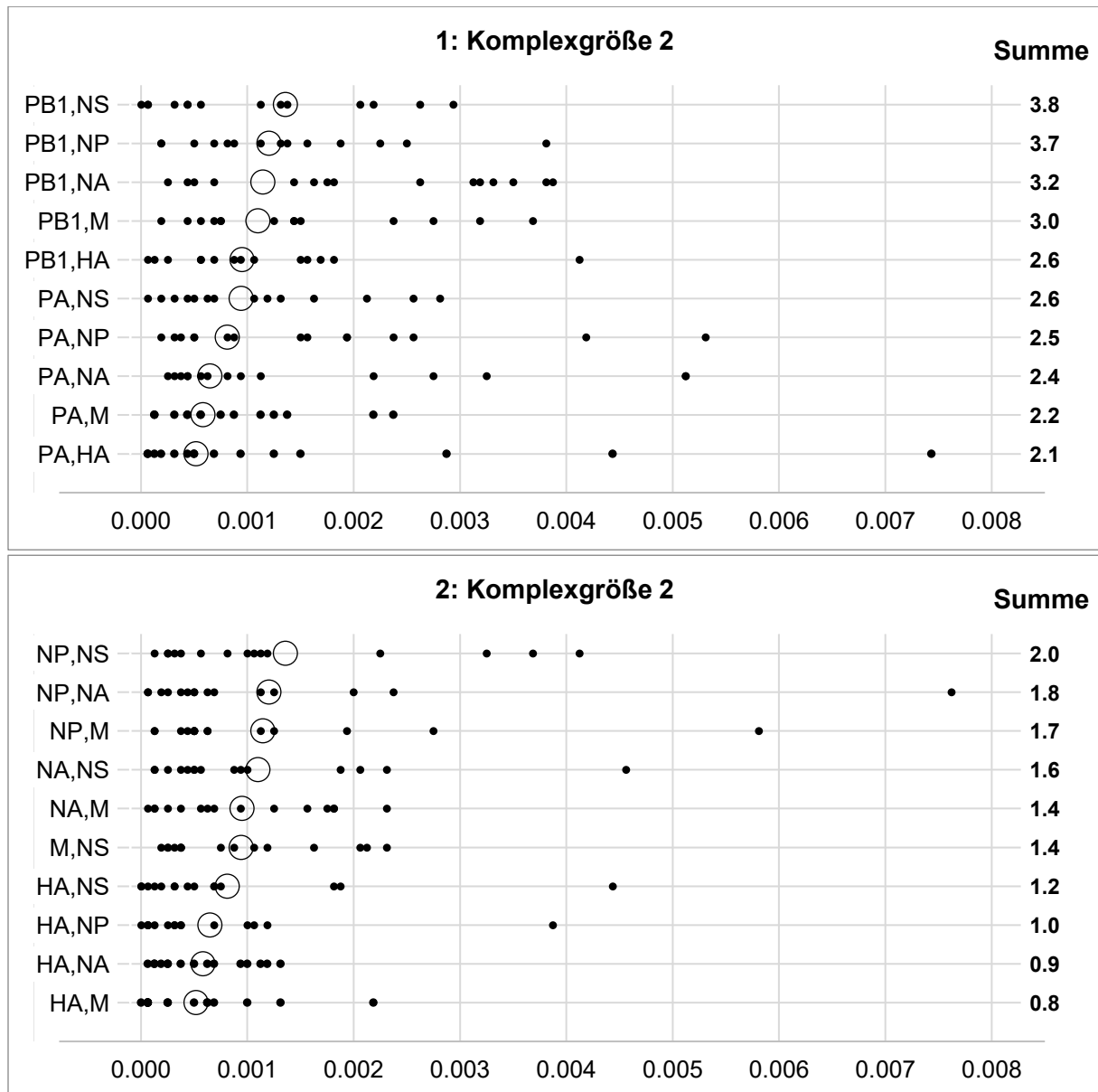
- van Rossum, Guido (2020b): The Python Library Reference, release 3.8.2. In: *Python Software Foundation*.
- Virtanen, Pauli; Gommers, Ralf; Oliphant, Travis E.; Haberland, Matt; Reddy, Tyler; Cournapeau, David et al. (2020): SciPy 1.0: fundamental algorithms for scientific computing in Python. In: *Nature methods* 17 (3), S. 261–272. DOI: 10.1038/s41592-019-0686-2.
- Wandzik, Joanna M.; Kouba, Tomas; Cusack, Stephen (2021): Structure and Function of Influenza Polymerase. In: *Cold Spring Harbor perspectives in medicine* 11 (9). DOI: 10.1101/cshperspect.a038372.
- WHO (Hg.) (2025): Influenza (seasonal). Online verfügbar unter [https://www.who.int/news-room/fact-sheets/detail/influenza-\(seasonal\)](https://www.who.int/news-room/fact-sheets/detail/influenza-(seasonal)), zuletzt aktualisiert am 28.02.2025, zuletzt geprüft am 24.06.2025.

## 7. Danksagung

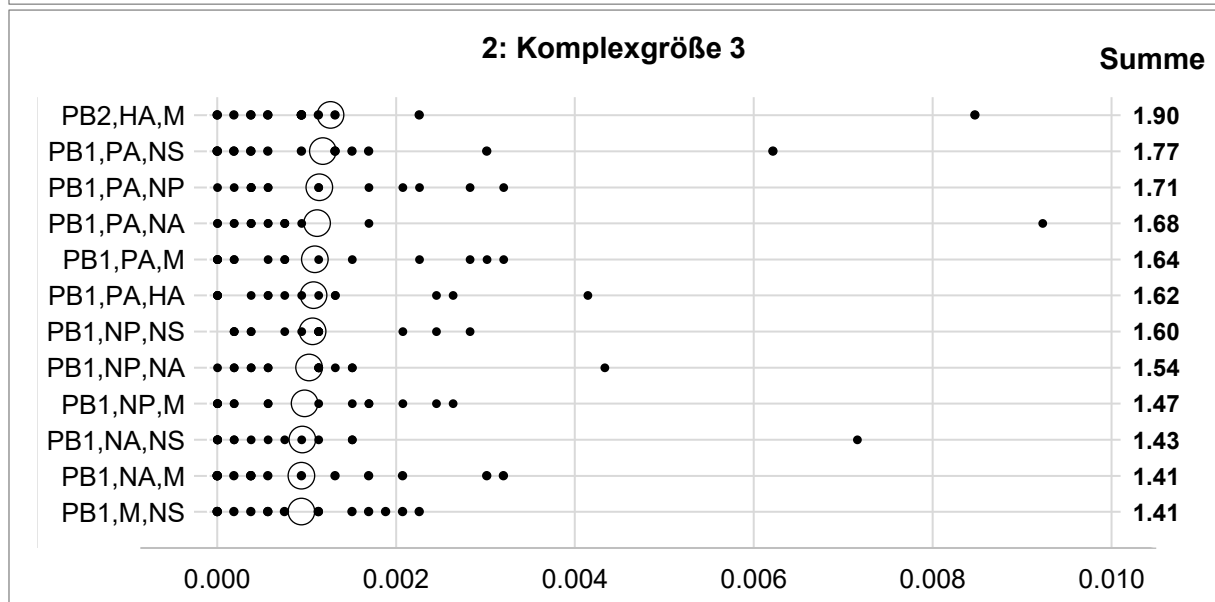
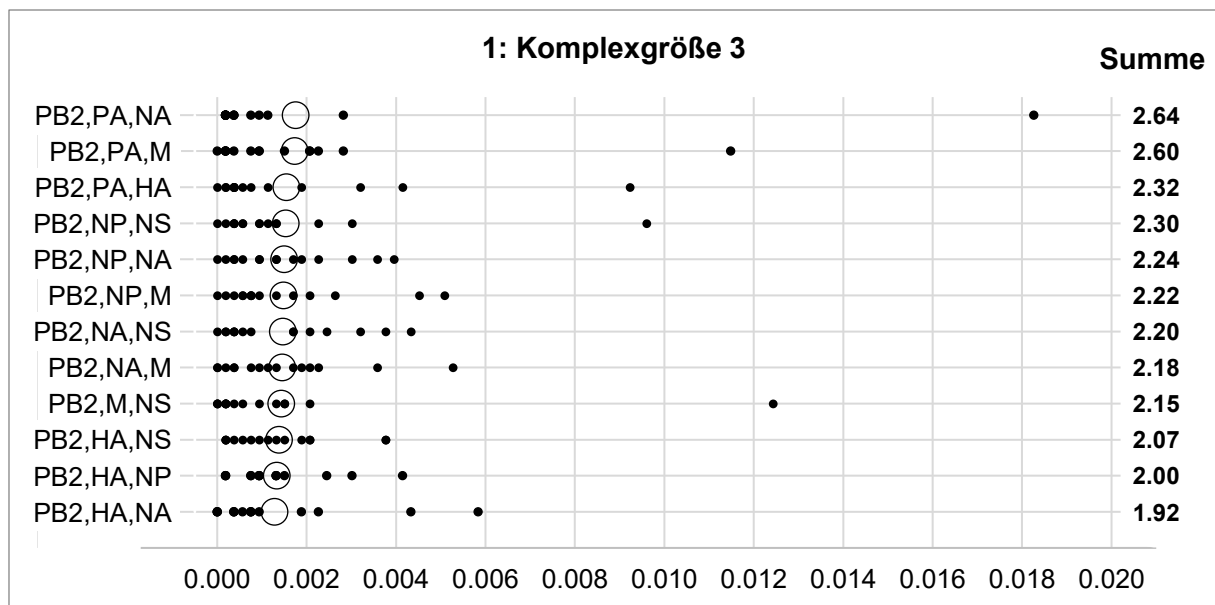
Ich möchte mich herzlich bei meinen Betreuern Dr. Markus Ankenbrand und Prof. Dr. Sabine Fischer für die hervorragende Betreuung dieser Arbeit bedanken. Mein besonderer Dank gilt Markus Ankenbrand für die stets schnelle und verlässliche Unterstützung bei fachlichen Fragen, sowie für die Einführung in das Feld der Bioinformatik und Bildverarbeitung. Ebenso danke ich den Kooperationspartnern Dr. Shazeb Ahmad und Dr. Redmond Smyth, die für die experimentelle Durchführung verantwortlich waren und diese Arbeit mit ihrer biologischen Expertise unterstützt haben. Nicht zuletzt danke ich meiner Familie und Freunden für ihre ständige Ermutigung und Geduld während meiner Arbeit.

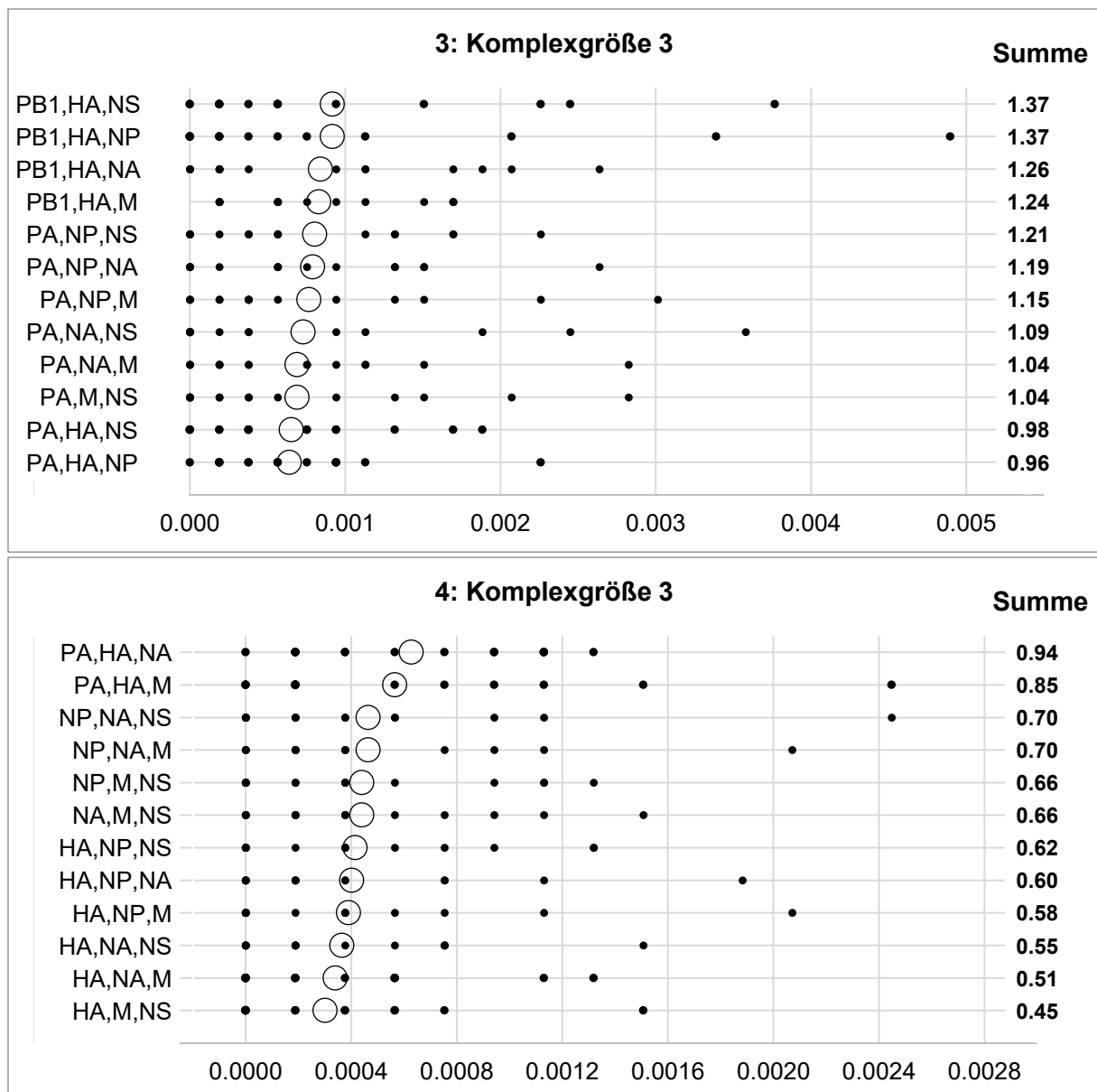
## 8. Anhang

### Anhang 1



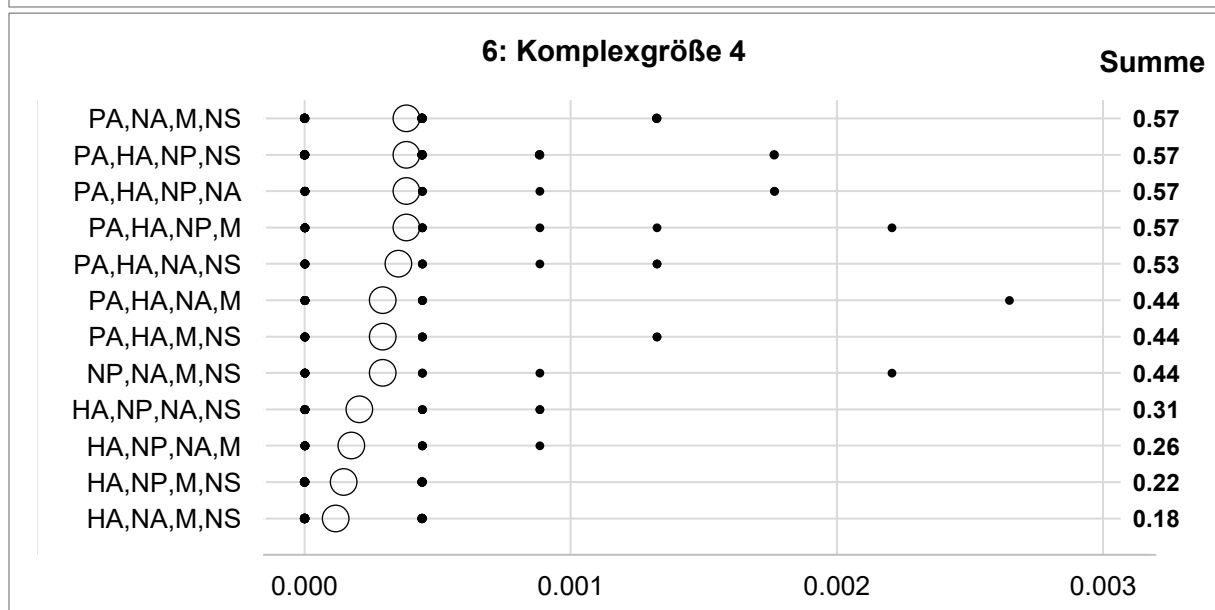
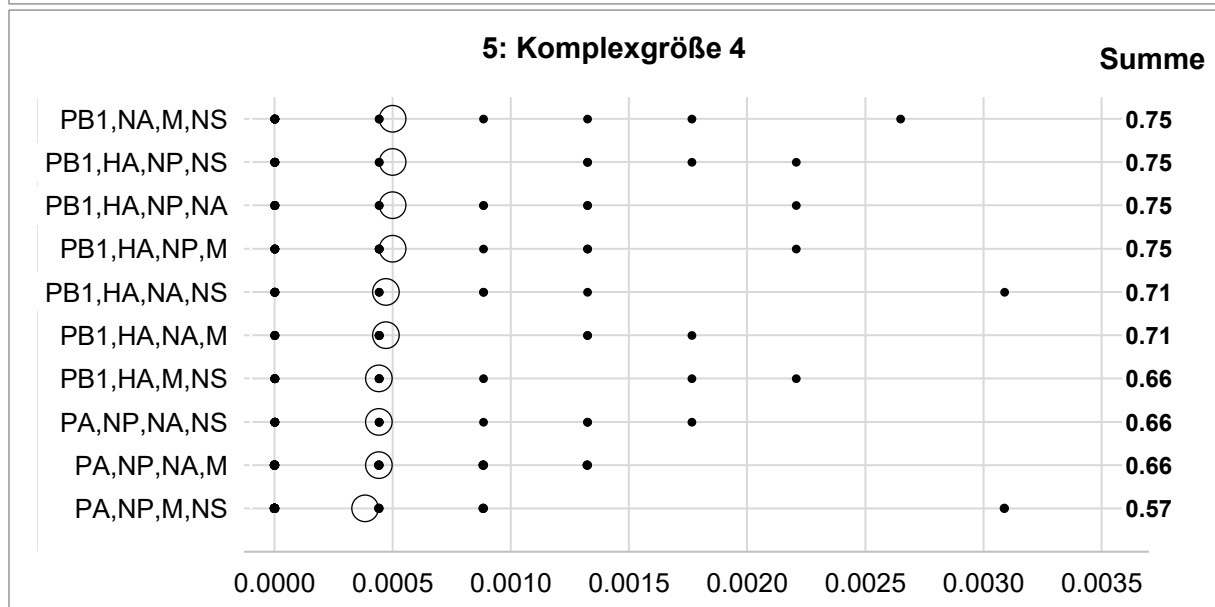
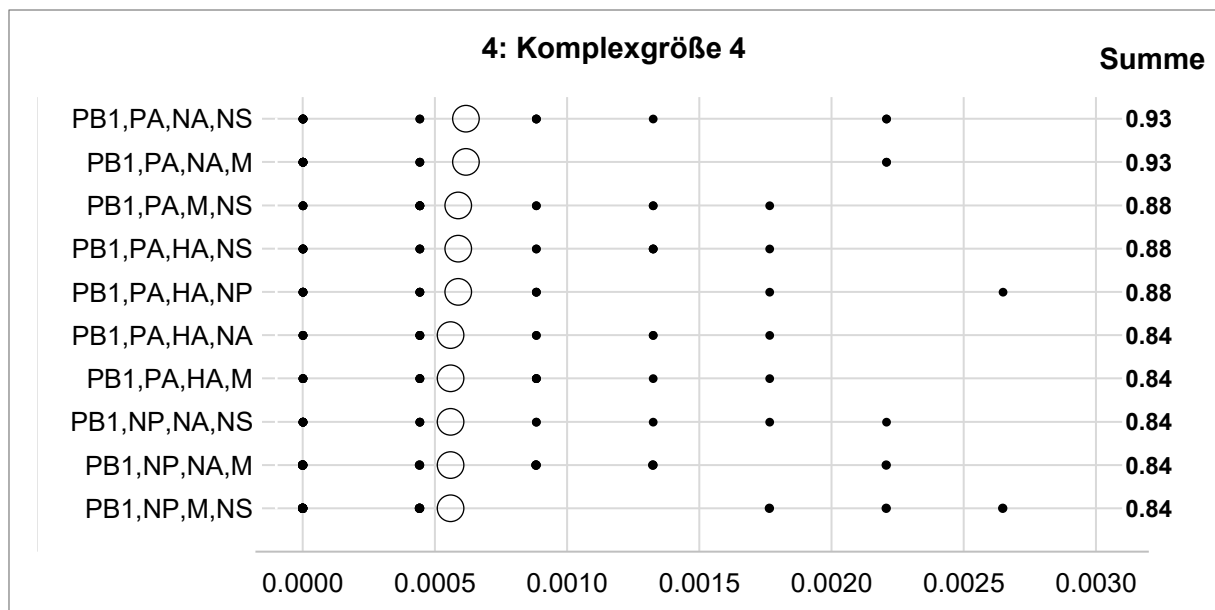
**Abb. 22: Relative Komplexzahlen der übrigen Zweierkomplexe der Infektionszeitpunkte vier bis sechs.** Die kleinen schwarzen Punkte zeigen die relativen Häufigkeiten der jeweils fünf Replikate der Infektionszeitpunkte vier bis sechs ( $n=15$ ) innerhalb der Komplexgröße. Die großen weißen Punkte zeigen das arithmetische Mittel. Die Summe rechts beschreibt die relative Anzahl des Komplexes an der Gesamtzahl aller Komplexe derselben Größe. Durch Überlagerung von Einzelwerten an der Nulllinie, sind nicht alle Einzelwerte erkennbar. Alle Komplexe wurden mittels Spot-basierter Dekodierung in undichten Regionen detektiert.



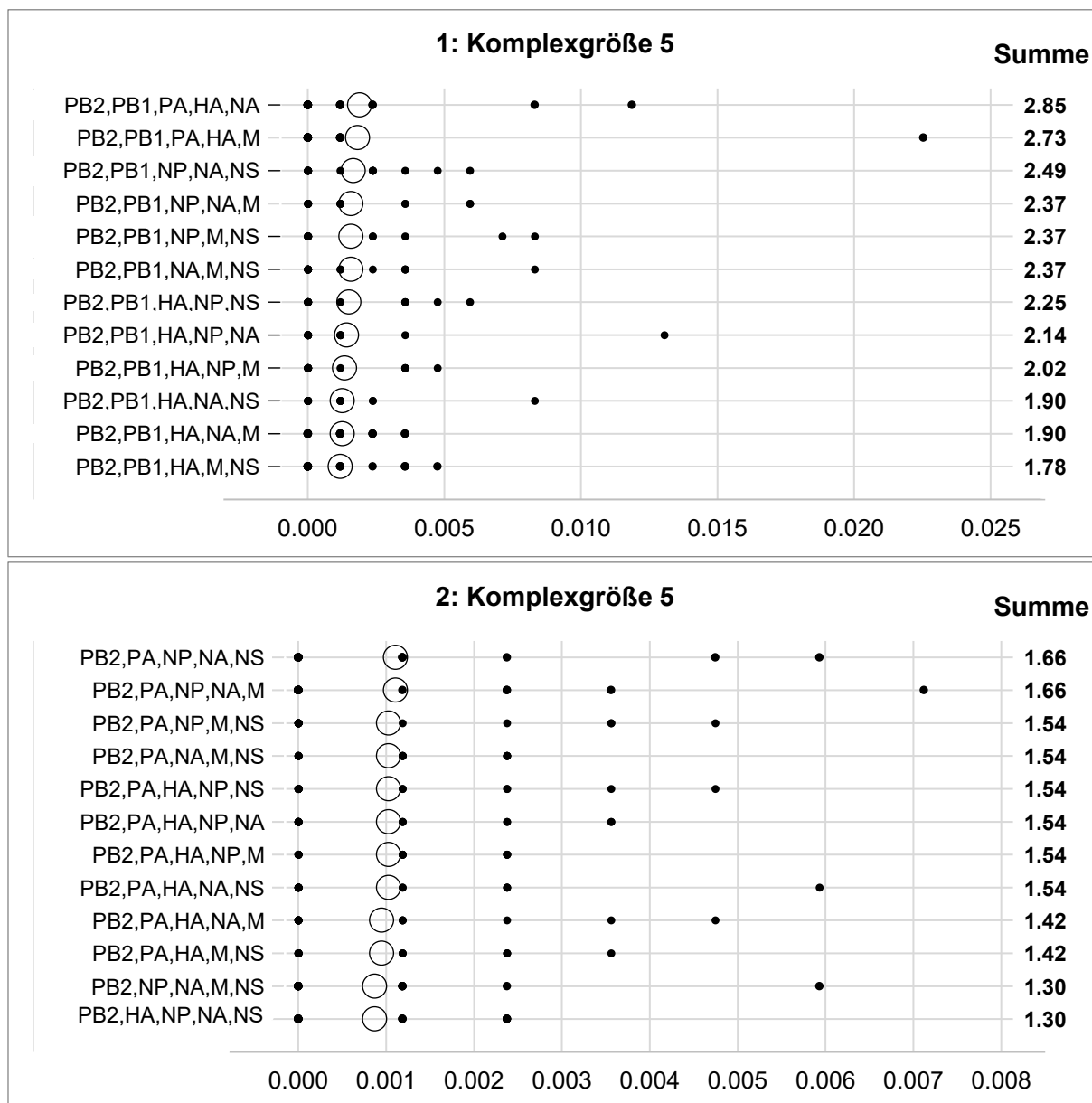


**Abb. 23: Relative Komplexzahlen der übrigen Dreierkomplexe der Infektionszeitpunkte vier bis sechs.** Die kleinen schwarzen Punkte zeigen die relativen Häufigkeiten der jeweils fünf Replikate der Infektionszeitpunkte vier bis sechs ( $n=15$ ) innerhalb der Komplexgröße. Die großen weißen Punkte zeigen das arithmetische Mittel. Die Summe rechts beschreibt die relative Anzahl des Komplexes an der Gesamtzahl aller Komplexe derselben Größe. Durch Überlagerung von Einzelwerten an der Nulllinie, sind nicht alle Einzelwerte erkennbar. Alle Komplexe wurden mittels Spot-basierter Dekodierung in undichten Regionen detektiert.

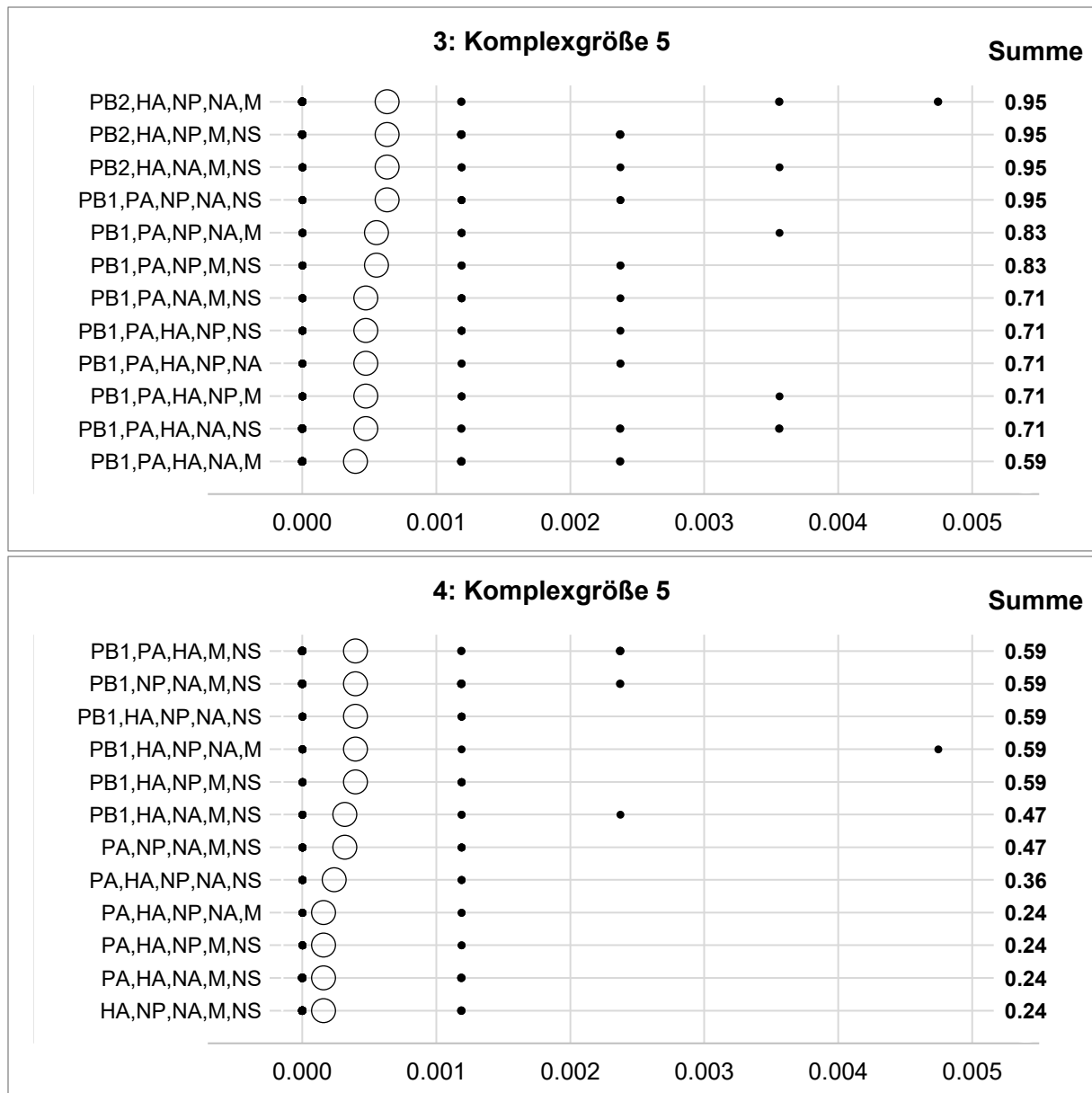




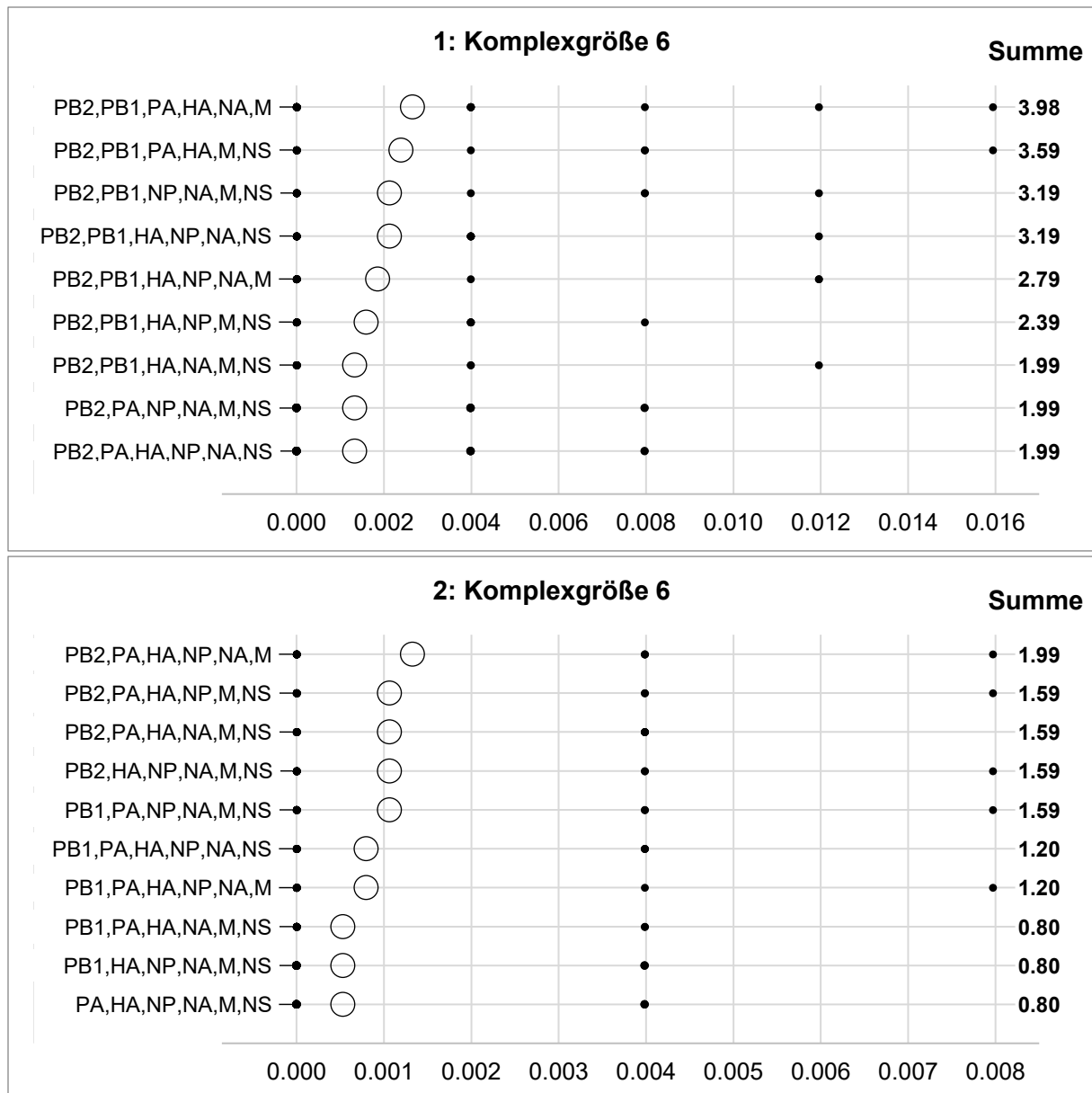
**Abb. 24: Relative Komplexzahlen der übrigen Viererkomplexe der Infektionszeitpunkte vier bis sechs.** Die kleinen schwarzen Punkte zeigen die relativen Häufigkeiten der jeweils fünf Replikate der Infektionszeitpunkte vier bis sechs ( $n=15$ ) innerhalb der Komplexgröße. Die großen weißen Punkte zeigen das arithmetische Mittel. Die Summe rechts beschreibt die relative Anzahl des Komplexes an der Gesamtzahl aller Komplexe derselben Größe. Durch Überlagerung von Einzelwerten an der Nulllinie, sind nicht alle Einzelwerte erkennbar. Alle Komplexe wurden mittels Spot-basierter Dekodierung in undichten Regionen detektiert.





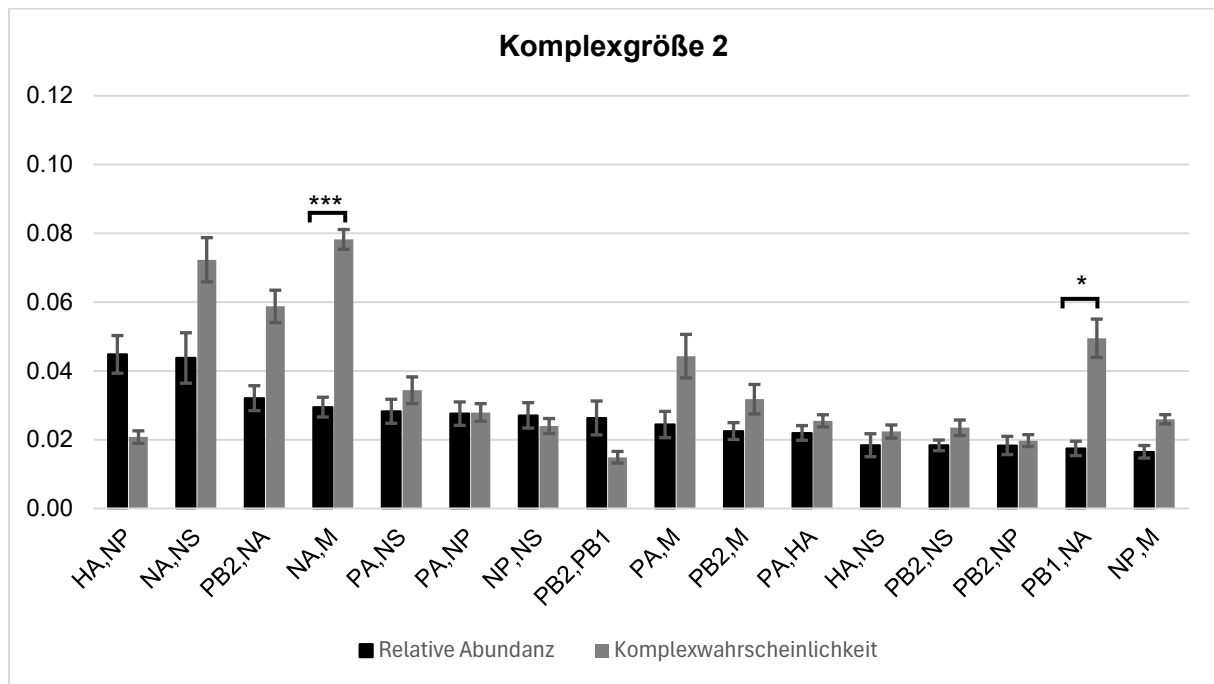


**Abb. 25: Relative Komplexzahlen der übrigen Fünferkomplexe der Infektionszeitpunkte vier bis sechs.** Die kleinen schwarzen Punkte zeigen die relativen Häufigkeiten der jeweils fünf Replikate der Infektionszeitpunkte vier bis sechs ( $n=15$ ) innerhalb der Komplexgröße. Die großen weißen Punkte zeigen das arithmetische Mittel. Die Summe rechts beschreibt die relative Anzahl des Komplexes an der Gesamtzahl aller Komplexe derselben Größe. Durch Überlagerung von Einzelwerten an der Nulllinie, sind nicht alle Einzelwerte erkennbar. Alle Komplexe wurden mittels Spot-basierter Dekodierung in undichten Regionen detektiert.

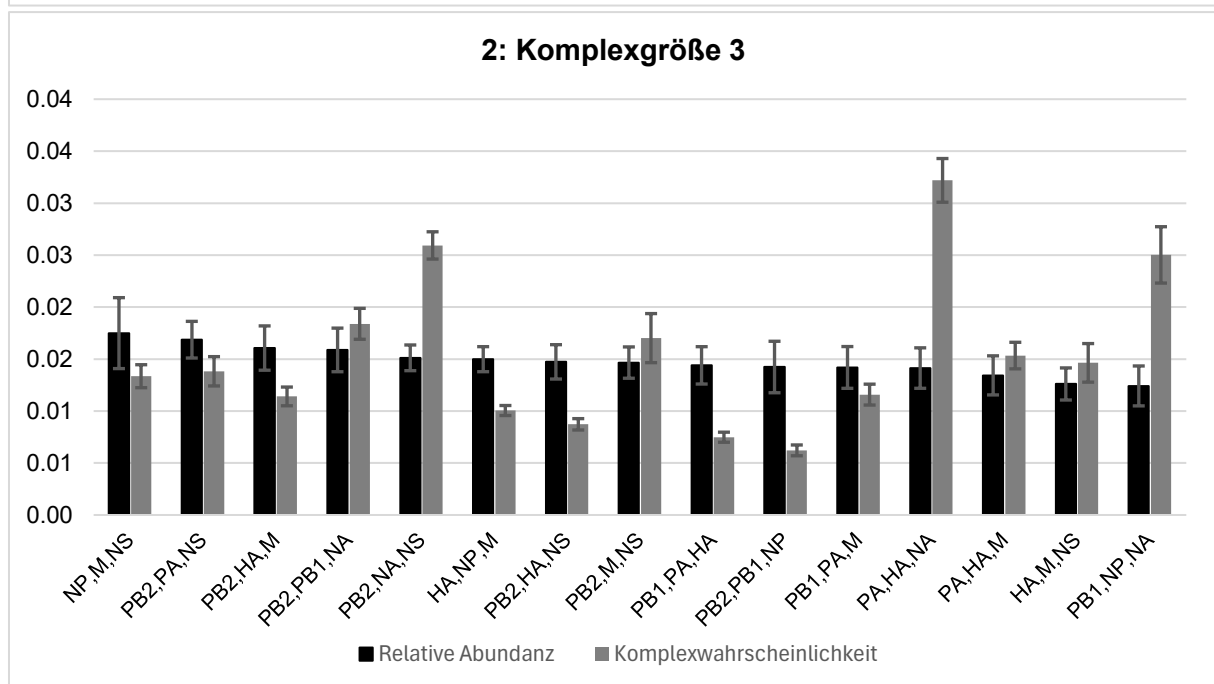
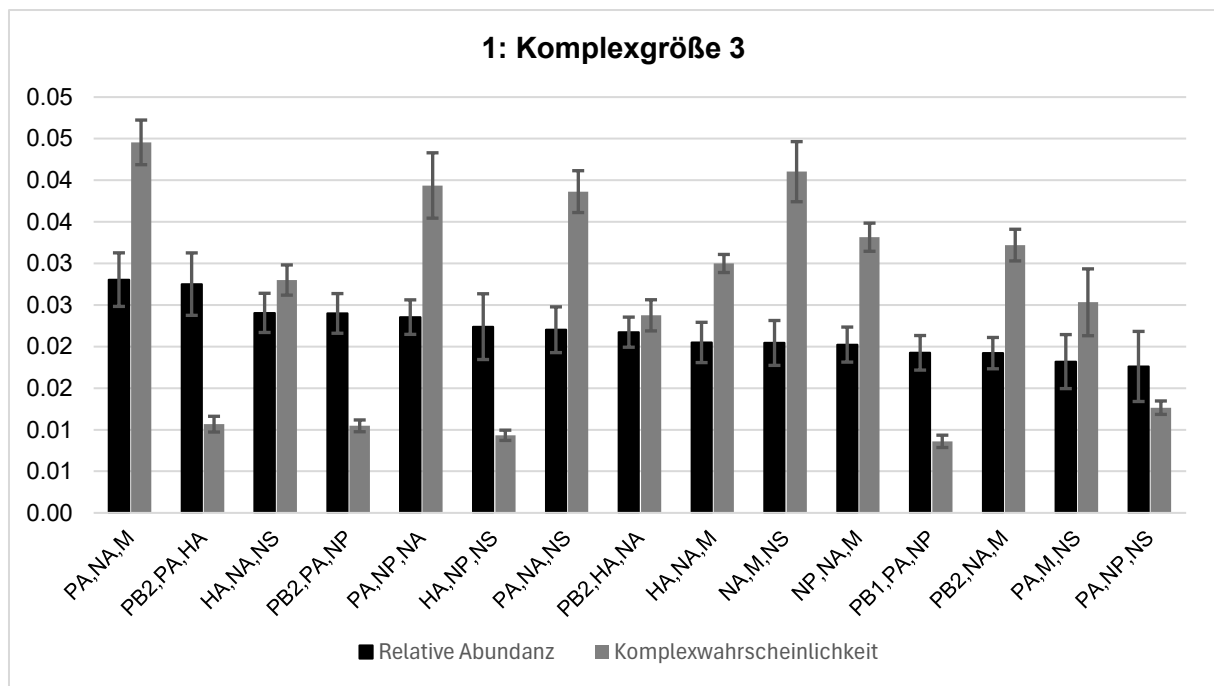


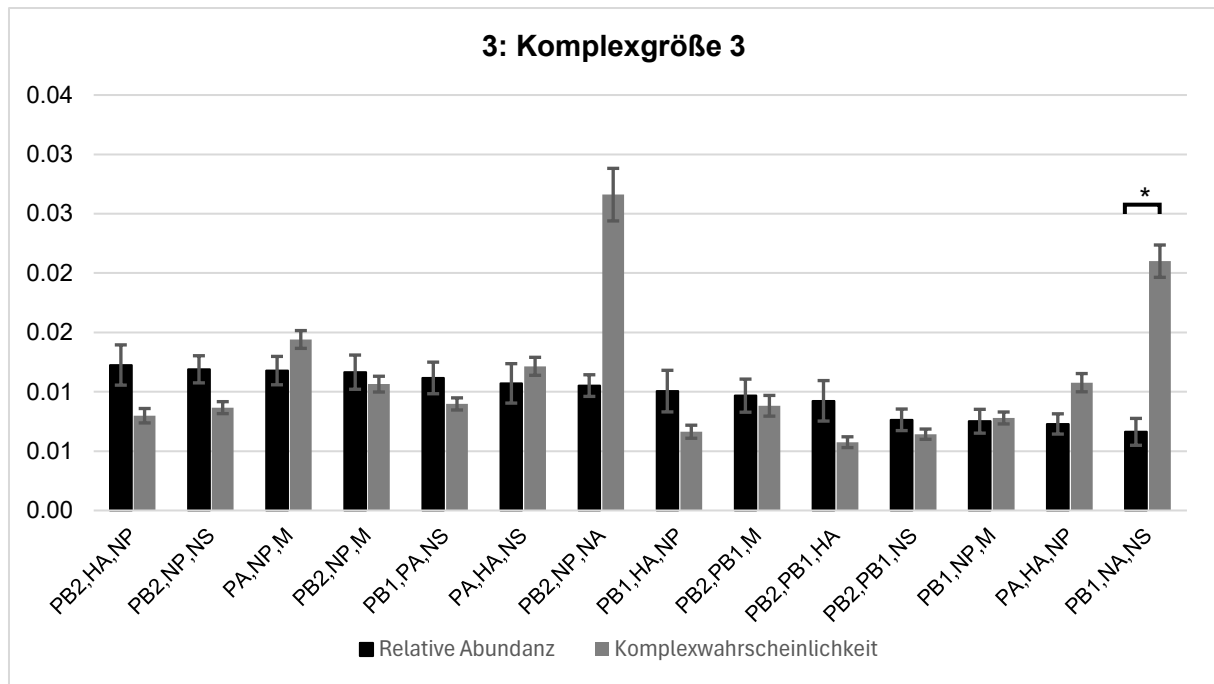
**Abb. 26: Relative Komplexzahlen der übrigen Sechserkomplexe der Infektionszeitpunkte vier bis sechs.** Die kleinen schwarzen Punkte zeigen die relativen Häufigkeiten der jeweils fünf Replikate der Infektionszeitpunkte vier bis sechs ( $n=15$ ) innerhalb der Komplexgröße. Die großen weißen Punkte zeigen das arithmetische Mittel. Die Summe rechts beschreibt die relative Anzahl des Komplexes an der Gesamtzahl aller Komplexe derselben Größe. Durch Überlagerung von Einzelwerten an der Nulllinie, sind nicht alle Einzelwerte erkennbar. Alle Komplexe wurden mittels Spot-basierter Dekodierung in undichten Regionen detektiert.

## Anhang 2

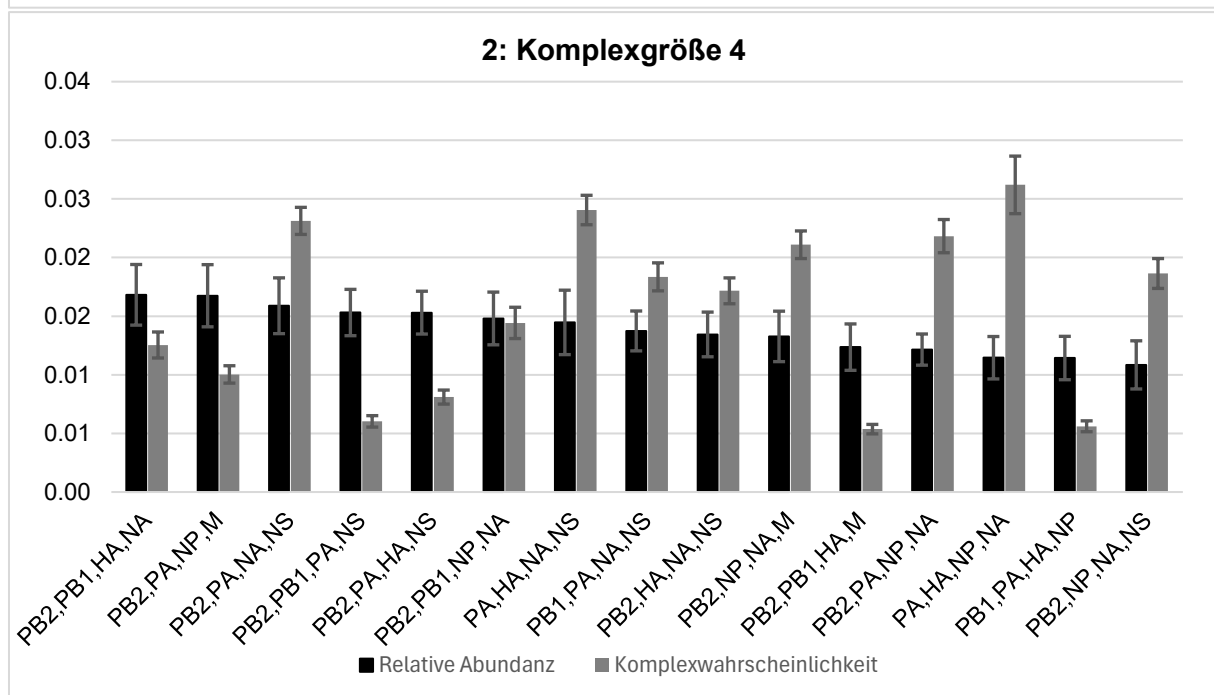
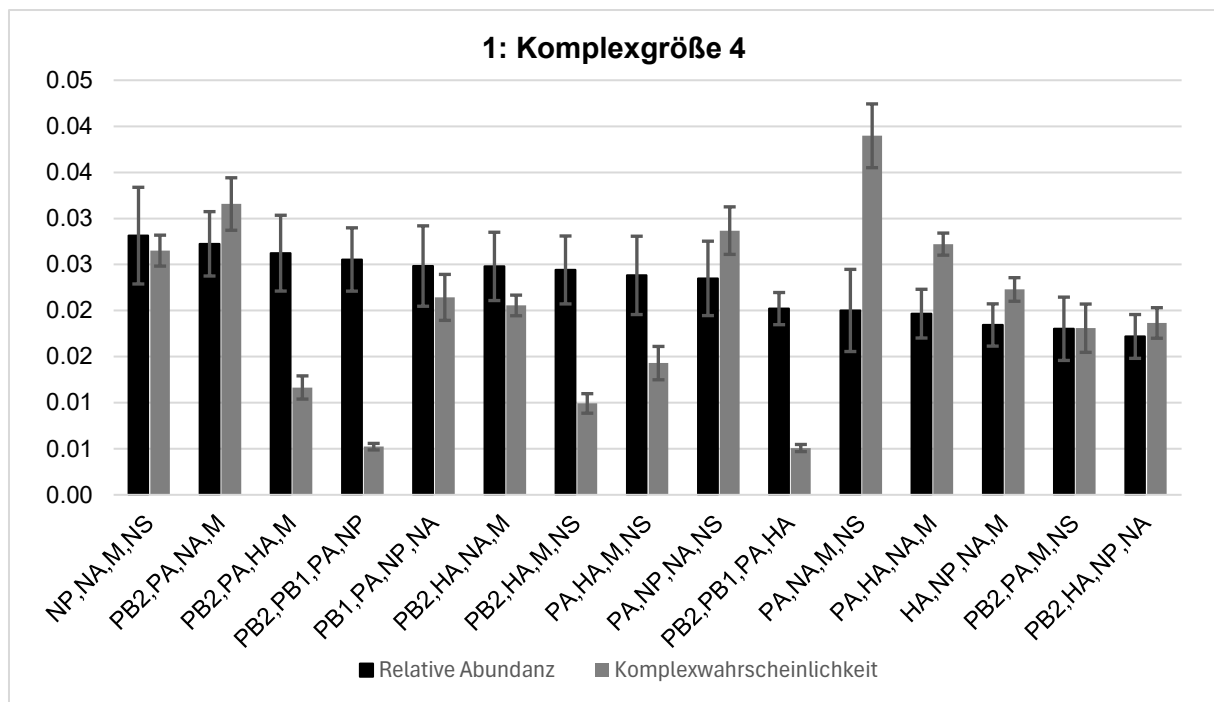


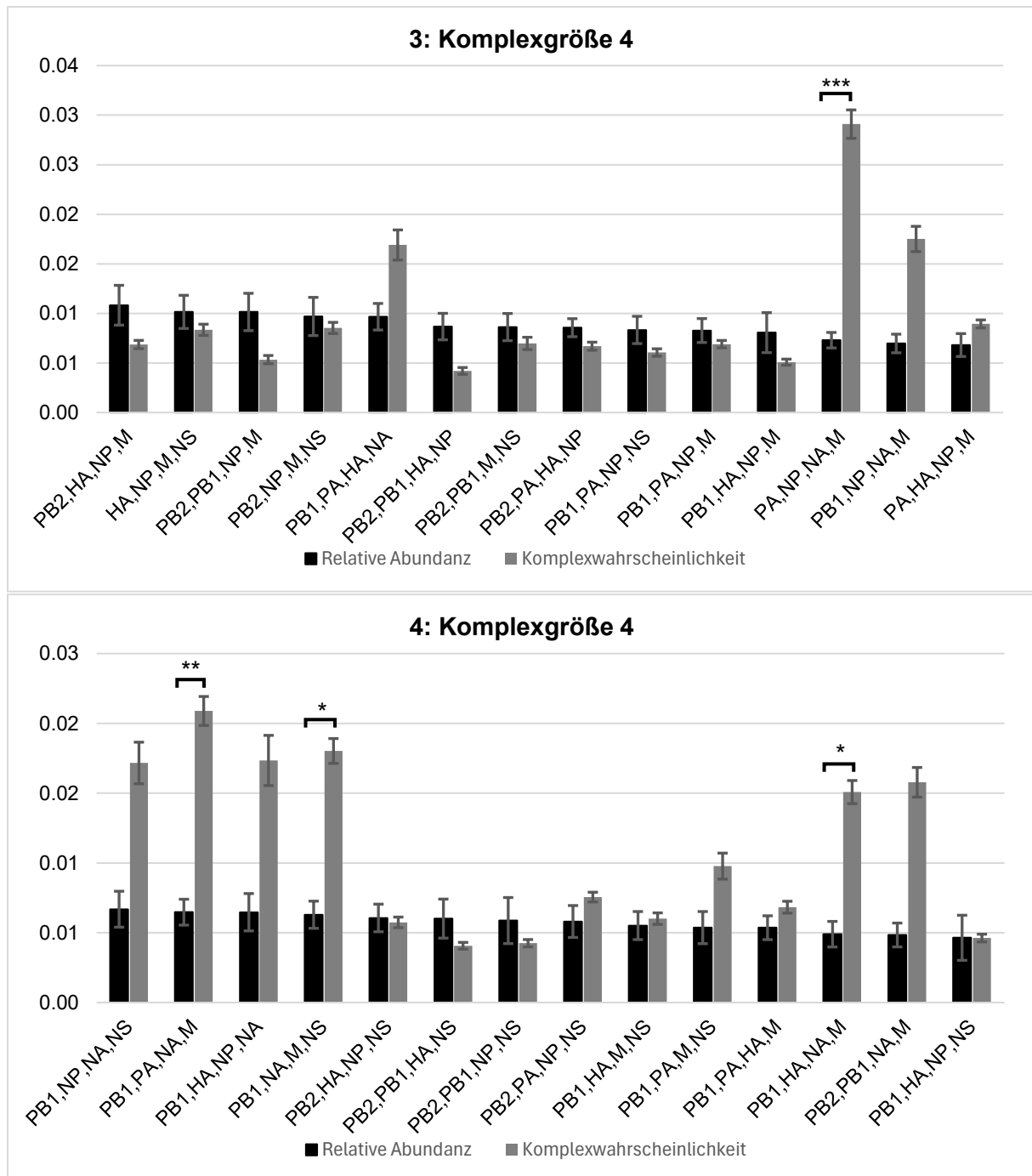
**Abb. 27: Vergleich von relativer Abundanz und Komplexwahrscheinlichkeit der übrigen Zweierkomplexe.** Die Komplexwahrscheinlichkeiten der Komplexe wurden durch Multiplikation der relativen Häufigkeiten der Einzelsegmente berechnet. Alle Einzelsegmente und Komplexe wurden durch Spot-basierte Dekodierung in undichten Regionen detektiert. Die Abbildung zeigt das arithmetische Mittel der relativen Abundanz (schwarz) und der Komplexwahrscheinlichkeit (grau) ( $n=15$ ) der fünf Replikate und Infektionszeitpunkte vier bis sechs. Die Fehlerbalken zeigen die Standardfehler. Sternchen markieren signifikante Unterschiede zwischen beobachteten relativen Abundanzen und berechneten Komplexwahrscheinlichkeiten ( $\alpha = 0,05$ ), ermittelt per unabhängigem, zweiseitigem T-Test. Die berechneten  $p$ -Werte wurden nach der Methode von Bonferroni für multiples Testen korrigiert. (Bland und Altman 1995). Ein Sternchen markiert einen  $p$ -Wert kleiner gleich 0,05 und drei Sternchen einen  $p$ -Wert kleiner gleich 0,001.





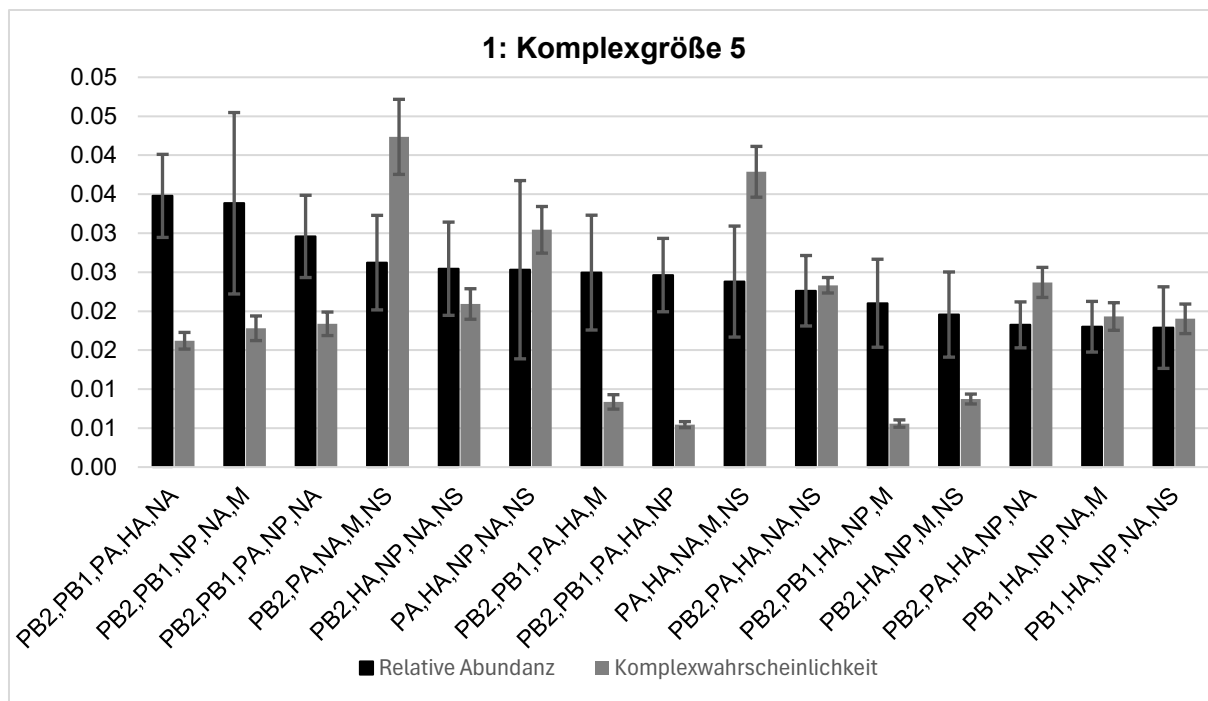
**Abb. 28 Vergleich von relativer Abundanz und Komplexwahrscheinlichkeit der übrigen Dreierkomplexe.** Die Komplexwahrscheinlichkeiten der Komplexe wurden durch Multiplikation der relativen Häufigkeiten der Einzelsegmente berechnet. Alle Einzelsegmente und Komplexe wurden durch Spot-basierte Dekodierung in undichten Regionen detektiert. Jede Abbildung zeigt das arithmetische Mittel der relativen Abundanz (schwarz) und der Komplexwahrscheinlichkeit (grau) ( $n=15$ ) der fünf Replikate und Infektionszeitpunkte vier bis sechs. (1), (2), (3) Dargestellt sind die jeweils 15 bzw. 14 relativen Abundanzen und Komplexwahrscheinlichkeiten von Dreierkomplexen, sortiert nach absteigender relativer Abundanz. Die Fehlerbalken zeigen die Standardfehler. Sternchen markieren signifikante Unterschiede zwischen beobachteten relativen Abundanzen und berechneten Komplexwahrscheinlichkeiten ( $\alpha = 0,05$ ), ermittelt per unabhängigem, zweiseitigem T-Test. Die berechneten p-Werte wurden nach der Methode von Bonferroni für multiples Testen korrigiert. (Bland und Altman 1995). Ein Sternchen markiert einen p-Wert kleiner gleich 0,05.



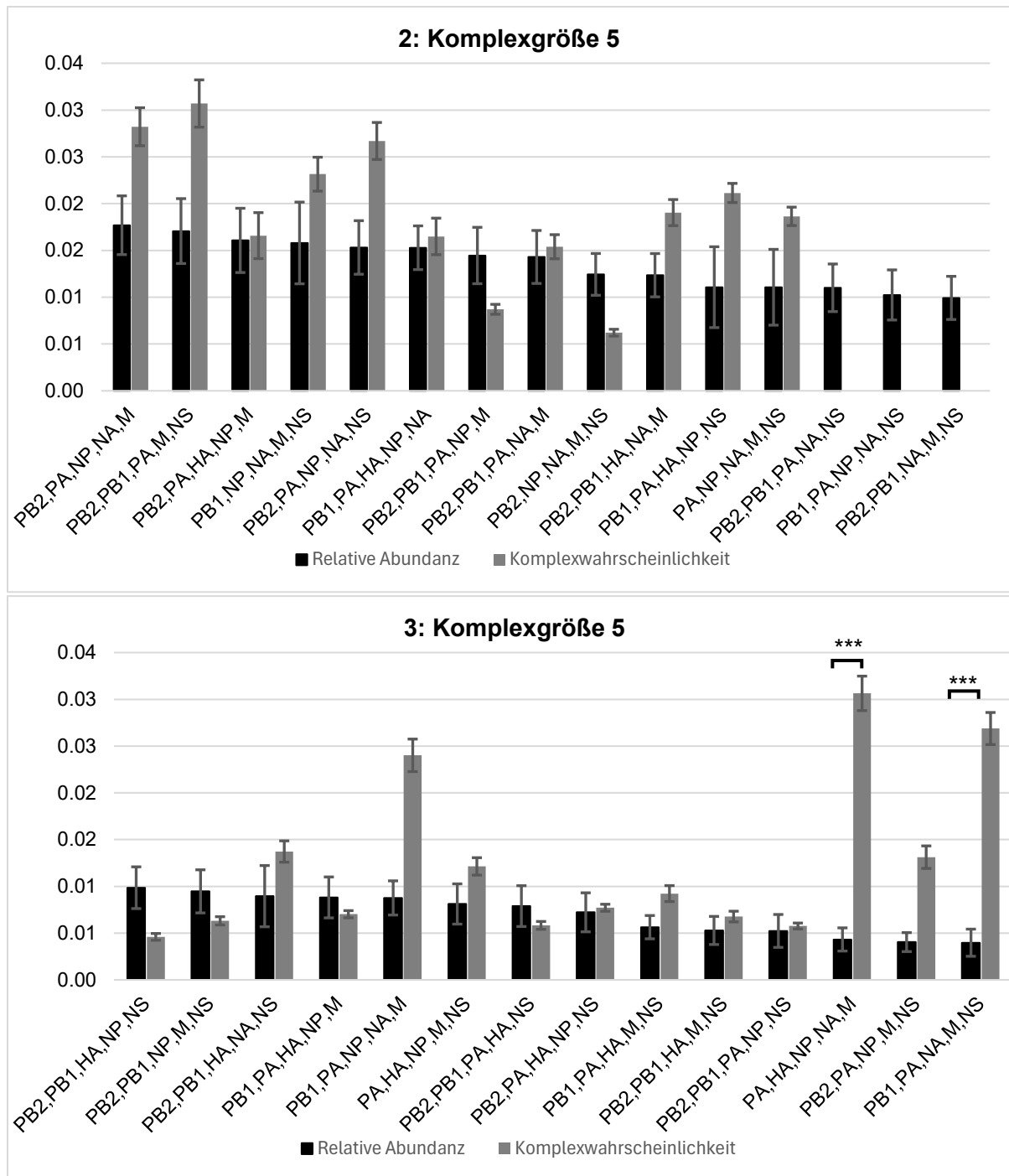


**Abb. 29: Vergleich von relativer Abundanz und Komplexwahrscheinlichkeit der übrigen Viererkomplexe.** Die Komplexwahrscheinlichkeiten der Komplexe wurden durch Multiplikation der relativen Häufigkeiten der Einzelsegmente berechnet. Alle Einzelsegmente und Komplexe wurden durch Spot-basierte Dekodierung in undichten Regionen detektiert. Jede Abbildung zeigt das arithmetische Mittel der relativen Abundanz (schwarz) und der Komplexwahrscheinlichkeit (grau) ( $n=15$ ) der fünf Replikate und Infektionszeitpunkte vier bis sechs. (1), (2), (3), (4) Dargestellt sind die jeweils 15 bzw. 14 relativen Abundanzen und Komplexwahrscheinlichkeiten von Viererkomplexen, sortiert nach absteigender relativer Abundanz. Die Fehlerbalken zeigen die Standardfehler. Sternchen markieren signifikante Unterschiede zwischen beobachteten relativen Abundanzen und berechneten Komplexwahrscheinlichkeiten ( $\alpha = 0,05$ ), ermittelt per unabhängigem, zweiseitigem T-Test. Die

berechneten  $p$ -Werte wurden nach der Methode von Bonferroni für multiples Testen korrigiert. (Bland und Altman 1995). Ein Sternchen markiert einen  $p$ -Wert kleiner gleich 0,05. Zwei Sternchen markieren einen  $p$ -Wert kleiner gleich 0,01 und drei Sternchen einen  $p$ -Wert kleiner gleich 0,001.

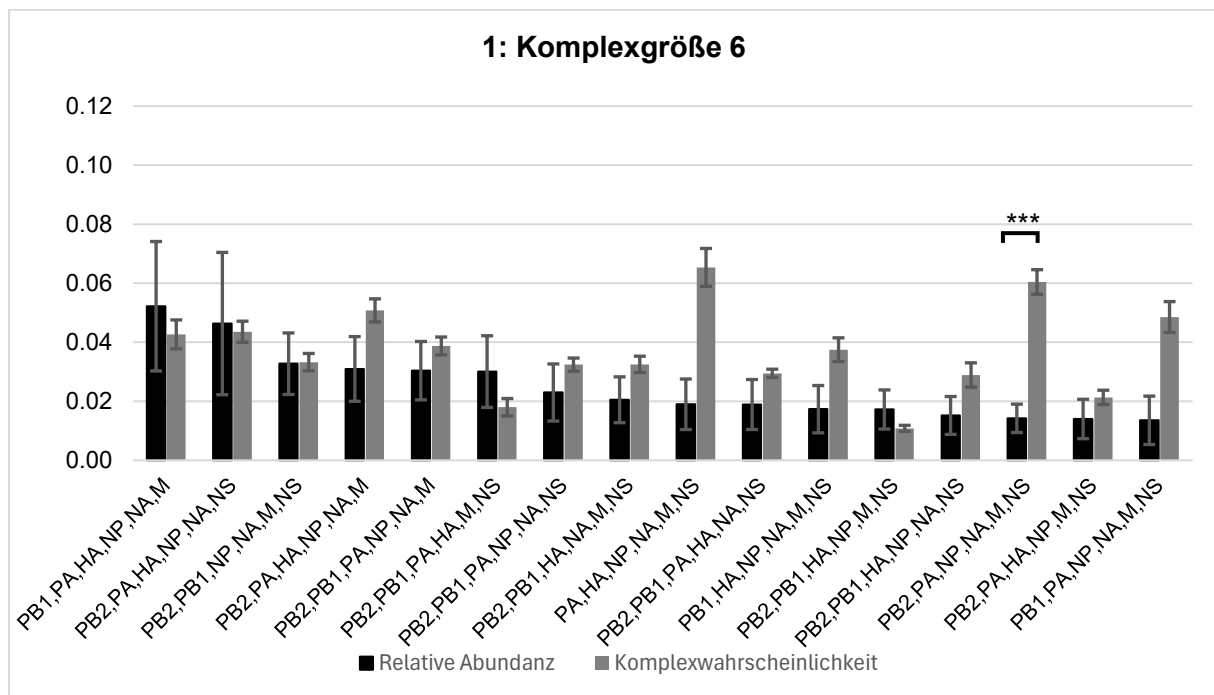




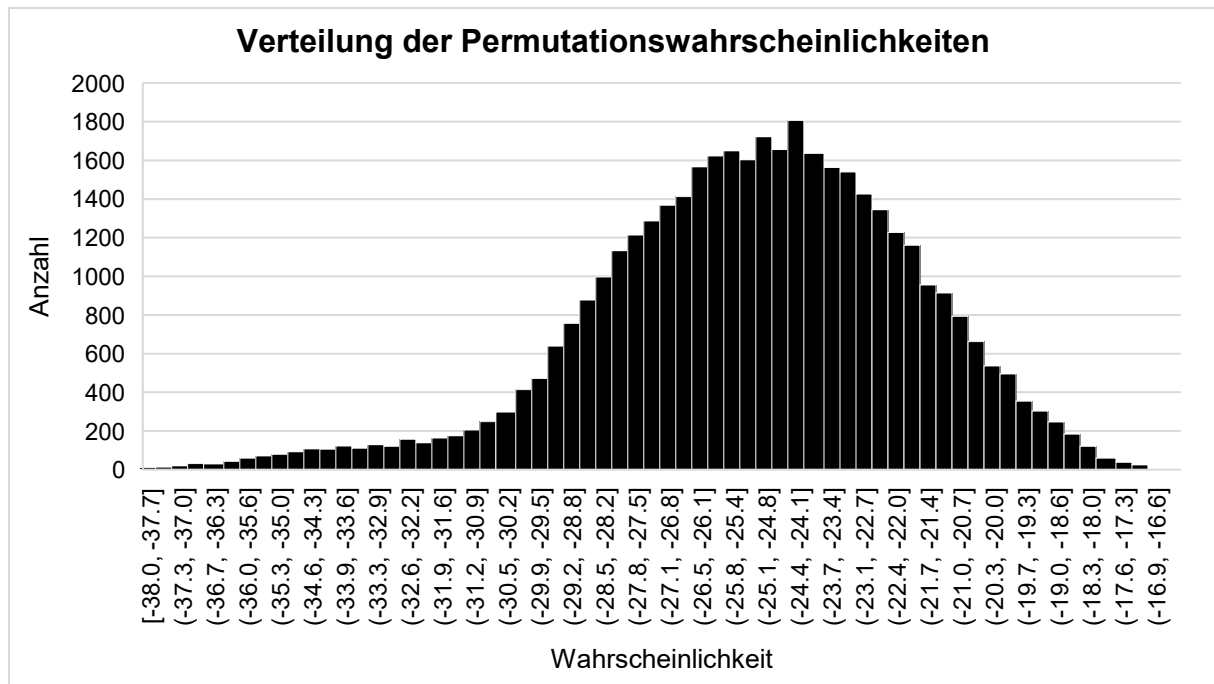


**Abb. 30: Vergleich von relativer Abundanz und Komplexwahrscheinlichkeit der übrigen Fünferkomplexe.** Die Komplexwahrscheinlichkeiten der Komplexe wurden durch Multiplikation der relativen Häufigkeiten der Einzelsegmente berechnet. Alle Einzelsegmente und Komplexe wurden durch Spot-basierte Dekodierung in undichten Regionen detektiert. Jede Abbildung zeigt das arithmetische Mittel der relativen Abundanz (schwarz) und der Komplexwahrscheinlichkeit (grau) ( $n=15$ ) der fünf Replikate und Infektionszeitpunkte vier bis sechs. (1), (2), (3) Dargestellt sind die jeweils 15 bzw. 14 relativen Abundanzen und Komplexwahrscheinlichkeiten von Viererkomplexen, sortiert nach absteigender relativer Abundanz. Die Fehlerbalken zeigen die Standardfehler. Sternchen markieren signifikante Unterschiede zwischen beobachteten relativen Abundanzen und berechneten Komplexwahrscheinlichkeiten ( $\alpha = 0,05$ ), ermittelt per unabhängigem, zweiseitigem T-Test. Die

berechneten p-Werte wurden nach der Methode von Bonferroni für multiples Testen korrigiert. (Bland und Altman 1995). Drei Sternchen markieren einen p-Wert kleiner gleich 0,001.



**Abb. 31 Vergleich von relativer Abundanz und Komplexwahrscheinlichkeit der übrigen Sechserkomplexe.** Die Komplexwahrscheinlichkeiten der Komplexe wurden durch Multiplikation der relativen Häufigkeiten der Einzelsegmente berechnet. Alle Einzelsegmente und Komplexe wurden durch Spot-basierte Dekodierung in undichten Regionen detektiert. Die Abbildung zeigt das arithmetische Mittel der relativen Abundanz (schwarz) und der Komplexwahrscheinlichkeit (grau) ( $n=15$ ) der fünf Replikate und Infektionszeitpunkte vier bis sechs. Die Fehlerbalken zeigen die Standardfehler. Sternchen markieren signifikante Unterschiede zwischen beobachteten relativen Abundanzen und berechneten Komplexwahrscheinlichkeiten ( $\alpha = 0,05$ ), ermittelt per unabhängigem, zweiseitigem T-Test. Die berechneten p-Werte wurden nach der Methode von Bonferroni für multiples Testen korrigiert. (Bland und Altman 1995). Drei Sternchen markieren einen p-Wert kleiner gleich 0,001.



**Abb. 32: Verteilung der Wahrscheinlichkeiten aller 40.320 möglichen Permutationen der Genomassemblierung.** Die Wahrscheinlichkeiten wurden durch Addition der logarithmierten relativen Häufigkeiten der intermediären Komplexe berechnet. Die Wahrscheinlichkeiten wurden basierend auf den relativen Häufigkeiten der Infektionszeitpunkte vier bis sechs, detektiert mittels Spot-basierter Dekodierung in undichten Regionen, berechnet. Die Säulen zeigen die Anzahl an Wahrscheinlichkeiten innerhalb eines bestimmten Intervalls.