

Algorithms for the Correction of Photobleaching

Rory Nolan

2018-06-29

Contents

Preface	5
Ways to read	5
Apportion of credit	5
Publications	7
1 Introduction	11
1.1 Confocal light microscopy	11
1.2 Intensity traces	12
1.3 Fluorescence fluctuation spectroscopy	14
1.4 Fluorescence correlation spectroscopy	18
2 Instrumentation and Software	27
2.1 Instrumentation	27
2.2 Software programs, languages and tools	28
2.3 My software packages	30

3	Photobleaching Correction	33
3.1	Introduction to photobleaching	33
3.2	The effects of bleaching in FCS and FFS	35
3.3	Exponential fitting detrending	38
3.4	Boxcar smoothing detrending	44
3.5	Choosing the correct smoothing parameter for detrending	47
3.6	Exponential smoothing detrending	49
3.7	Correcting for non-stationary variance	52
3.8	Caveats of fitting and smoothing approaches to detrending	53
3.9	Robin Hood detrending	55
3.10	A comparison of detrending methods	58
4	Applications	63
4.1	Dimerization of FKBP12	63
4.2	<i>In vitro</i> number and brightness	64
4.3	HIV-1 receptor stoichiometry	66
5	Future plans	75
5.1	Robin Hood publication	75
5.2	Translate software to <i>ImageJ</i>	75

Preface

Ways to read

In my opinion, the best way to read this thesis is on the web at <https://rorynolan.github.io/phdthesis/>. If you are reading this on the web now but would like a pdf version, click on the download symbol at the top left of the page (to the right of the **A**) and select PDF.

Apportion of credit

Whilst I have attempted to make it clear throughout what I myself did and did not do in relation to all work mentioned, for clarity I include the following reliable rules of thumb:

- The idea for fluorescence fluctuation spectroscopy (FFS) as the main area of my thesis was Sergi Padilla-Parra's.
- Molecular biology was done by Maro Iliopoulou and Luis Alvarez.
- Imaging was done by Maro Iliopoulou, Luis Alvarez and Sergi Padilla-Parra.
- The idea that correction for bleaching was the crucial step for FFS analysis was formulated by Luis Alvarez, Sergi Padilla-Parra and I.
- I formulated the solutions for how to correctly correct for bleaching, i.e. the automatic parameter choice and the Robin Hood algorithm.
- I wrote all of the software and maintain all of it.
- All FFS analysis was performed using my software. The software was used to analyse data by Maro Iliopoulou, Luis Alvarez, Sergi Padilla Parra and I.
- Structural modelling was done by Thomas Bowden and Yasunori Watanabe.
- On all papers where I am the first listed author, I wrote the paper, taking suggested amendments and augmentation from other listed authors. The NSMB paper ([Iliopoulou et al., 2018](#)) was written by Maro and Sergi; I also made significant contributions to the writing of that manuscript but my main role in that project was in data analysis. In all other papers where I am listed as an author, my role was minor.

Sergi Padilla-Parra is my supervisor. Luis Alvarez is a postdoctoral researcher in

Sergi's group and Maro Iliopoulou is a research assistant in Sergi's group.

Publications

I hereby list my publications. It is common to do this at the end of a thesis, but I will be referring to these papers frequently, so I think it useful to highlight them here at the start. These can all be downloaded from <https://github.com/rorynolan/phdthesis/tree/master/papers>, where the naming convention is `JournalYEAR.pdf`.

First author

- R. Nolan and S. Padilla-Parra. “filesstrings: An R package for file and string manipulation”. In: *Journal of Open Source Software* 2.14 (2017).
- R. Nolan and S. Padilla-Parra. “exampletestr—An easy start to unit testing R packages”. In: *Wellcome open research* 2 (2017).
- R. Nolan, L. Alvarez, J. Elegheert, et al. “nandb—number and brightness in R with a novel automic detrending algorithm”. In: *Bioinformatics* 33.21 (2017).
- R. Nolan, M. Iliopoulou, L. Alvarez, et al. “Detecting protein aggregation

and interactions in live cells: a guide to Number and Brightness”. In: *Methods* (2017).

- R. Nolan and S. Padilla-Parra. “ijttiff: An R package providing TIFF I/O for ImageJ users”. In: *Journal of Open Source Software* 3.23 (2018).
- R. Nolan, L. Alvarez, S. C. Griffiths, et al. “Calibration-Free In-Vitro Quantification of Protein Homo-Oligomerization Using Commercial Instrumentation and Free, Open Source Brightness Analysis Software”. In: *Journal of Visualized Experiments* 0.0 (2018).

Co-first author

- M. Iliopoulou, R. Nolan, L. Alvarez, et al. “A dynamic three step mechanism drives the HIV-1 prefusion reaction”. In: *Nat. Struct. Mol. Biol.* 0.0 (2018).

Other

- D. M. Jones, L. A. Alvarez, R. Nolan, et al. “Dynamin-2 stabilizes the HIV-1 fusion pore with a low oligomeric state”. In: *Cell reports* 18.2 (2017).
- G. M. Jakobsdottir, M. Iliopoulou, R. Nolan, et al. “On the whereabouts of HIV-1 cellular entry and its fusion ports”. In: *Trends in molecular medicine* (2017).

- Q. F. Wills, E. Mellado-Gomez, R. Nolan, et al. “The nature and nurture of cell heterogeneity: accounting for macrophage gene-environment interactions with single-cell RNA-Seq”. In: *BMC genomics* 18.1 (2017).

Chapter 1

Introduction

1.1 Confocal light microscopy

All of the images used in my PhD were collected on a confocal microscope. This type of microscope guarantees that only in-focus light is collected at the detector. See figure [1.1](#).

Definition 1.1. The *confocal volume* is the *in-focus* volume within a sample that is efficiently detected using a system designed with confocal optics.^{[3](#)}

An image is acquired on a confocal microscope by scanning this apparatus across a sample, collecting one pixel at a time. Some confocal detectors collect intensity

³<http://www.fcsxpert.com/classroom/theory/what-is-confocal-volume.html>

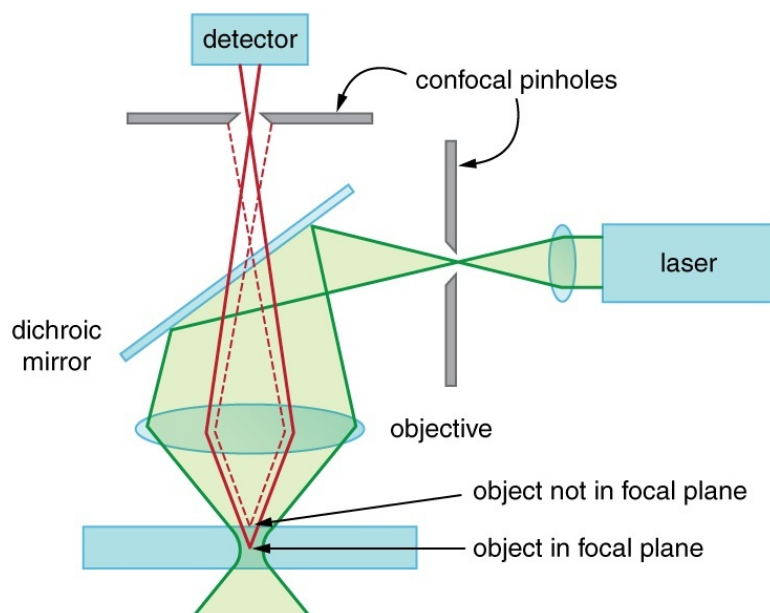


Figure 1.1: Confocal microscope light path showing how out of focus light does not make it to the detector.²

in arbitrary units but others are capable of photon-counting. The work in this thesis was carried out with photon-counting detectors.

Remark. Henceforth, intensity counts will be assumed to be in units of photons.

1.2 Intensity traces

Definition 1.2. A *fluorophore* is a fluorescent chemical compound that can re-emit light upon light excitation.⁴

Fluorophores under constant excitation emit light one photon at a time according

⁴<https://en.wikipedia.org/wiki/Fluorophore>

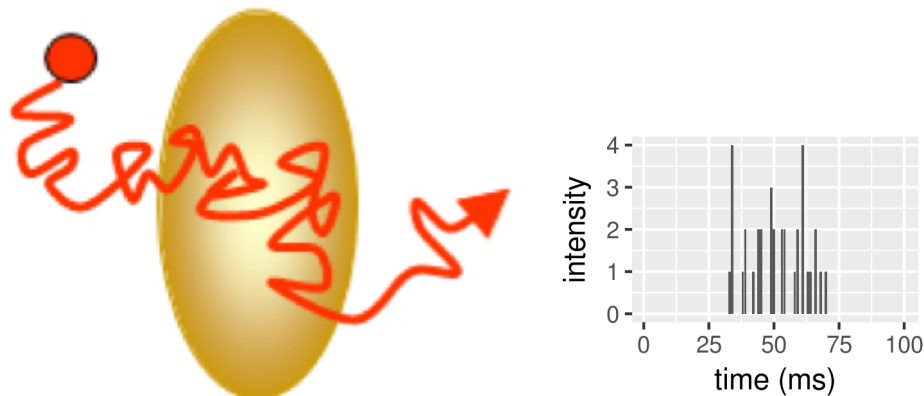


Figure 1.2: Left: a fluorophore diffusing through the confocal volume. Right: the intensity trace due to this fluorophore.

to poisson statistics.⁵

Consider figure 1.2. A fluorophore enters the confocal volume, is excited there and emits photons which are collected by the detector. When the fluorophore is not in the confocal volume (start and end), no photons are detected. When the particle is in the confocal volume, photons it emits are collected at the detector. Different numbers of photons are collected per unit time (per ms here).

Definition 1.3. The time-series of intensity counts in a given pixel or image is referred to as its *intensity trace*.

⁵https://en.wikipedia.org/wiki/Poisson_distribution

1.3 Fluorescence fluctuation spectroscopy

Broadly, fluorescence fluctuation spectroscopy (FFS) is the analysis of the intensity fluctuation of a fluorescence signal (Chen et al., 1999). This very often takes the form of moment analysis⁶ (Qian and Elson, 1990). Briefly, moment analysis is an attempt to extract data from a distribution of values using its *moments*. The first moment of a distribution is its mean value, the second moment is its variance and so on. Intensity traces can be viewed as distributions with moments. For example, the intensity trace in figure 1.2 has mean 0.35 and variance 0.69.

1.3.1 Number and brightness

Number and brightness (N&B, Digman et al. (2008)) is an FFS technique for quantifying the oligomeric states of fluorescently labelled proteins. What follows is a mathematical description of the technique.

Definition 1.4. An *entity* is a set of molecules which are chemically bound.

Definition 1.5. The brightness ϵ of an entity is the mean number of photon detector counts it gives per unit time when in the illumination (confocal) volume.

For an image series where the i^{th} slice in the stack is the image acquired at time

⁶[https://en.wikipedia.org/wiki/Method_of_moments_\(statistics\)](https://en.wikipedia.org/wiki/Method_of_moments_(statistics))

$t = i$, for a given pixel position (x, y) , we define $\langle I \rangle$ as the mean intensity of that pixel over the image series and σ^2 as the variance in that intensity. Define n as the mean number of entities in the illumination volume corresponding to that pixel. Assuming that all entities are mobile, we have

$$N = \frac{\langle I \rangle^2}{\sigma^2} = \frac{\epsilon n}{1 + \epsilon} \quad (1.1)$$

$$B = \frac{\sigma^2}{\langle I \rangle} = 1 + \epsilon \quad (1.2)$$

where N and B are referred to as the *apparent number* and *apparent brightness* respectively. This gives

$$\epsilon = \frac{\sigma^2}{\langle I \rangle} - 1 \quad (1.3)$$

$$n = \frac{\langle I \rangle^2}{\sigma^2 - \langle I \rangle} \quad (1.4)$$

— [Nolan et al. \(2017b\)](#)⁷

The quantity ϵ is a relative measure of oligomeric state. That is, ϵ will be twice

⁷This is a formulation of N&B that I wrote in a 2017 review of the technique.

as big for dimers as it is for monomers, three times as big for trimers as it is for monomers and so on.

The way that N&B experiments to determine unknown oligomeric states are generally done is as follows:

1. For a given laser power and fluorophore with a system where all entities are known to be monomeric, measure the brightness ϵ . Call this $\epsilon_{\text{monomer}}$.
2. With the same laser power and fluorophore but now with a system where the oligomeric state is unknown, measure the brightness ϵ again. Call this $\epsilon_{\text{unknown}}$.
3. The unknown oligomeric state is equal to $\epsilon_{\text{unknown}}/\epsilon_{\text{monomer}}$.

Number and brightness has specific pixel dwell-time and frame rate requirements.

These were first articulated in my 2017 review of N&B.

Definition 1.6. The *pixel dwell time* t_{dwell} of a scanning confocal microscope is the amount of time it spends collecting photons at each given pixel before moving on to the next pixel.

Definition 1.7. The *frame time* of a scanning confocal microscope is the amount of time it takes to acquire the data for all of the pixels in a whole frame. That is the length of time from the start of detection of photons from the first pixel to the end of detection of photons from the last pixel.

Definition 1.8. The mean *residence time* τ_D of a particle in the confocal volume is the average length of time that a particle that enters the confocal volume resides in that volume before exiting.

The requirement is $t_{\text{dwell}} \ll \tau_D \ll t_{\text{frame}}$. This ensures that:

1. When acquiring photons at a given pixel, the underlying configuration of entities is constant (there's not enough time for the entities to move and change their configuration).
2. When the scanner returns to a given pixel (one frame time later), the underlying configuration has changed totally since the last time the scanner was at this pixel (because so much time has passed, all of the diffusing entities have moved a lot in the meantime).

Both of these points are implicitly assumed in the derivation of the N&B equations so it is essential to get these acquisition parameters right. This is discussed at length in [Nolan et al. \(2017b\)](#).

An important property of N&B is that if all fluorescent particles are immobile, then $B = 1$. This is because photon emission from stationary sources happens according to a poisson distribution. Poisson distributions have variance equal to mean, this implies $\sigma^2 = \langle I \rangle$ which gives $B = \frac{\sigma^2}{\langle I \rangle} = 1$.

The N&B technique is fraught with technical difficulties. Principal of these is the problem of photobleaching. Most of my PhD focussed on corrections for photobleaching. This is discussed in chapter 3.

1.4 Fluorescence correlation spectroscopy

Fluorescence correlation spectroscopy (FCS) is the correlation analysis of fluorescence intensity fluctuations.⁸ For this reason, FCS can be described as a subfield of FFS (Jameson et al., 2009). In practice, FFS is mostly used to refer to the non-FCS parts of the whole FFS field. I will follow that convention.

First, let us introduce some concepts from statistics.

Definition 1.9 (statistics). The *correlation* between two random variables X and Y with expected values μ_X and μ_Y and standard deviations σ_X and σ_Y is

$$\text{corr}(X, Y) = \frac{E[(X - \mu_X)(Y - \mu_Y)]}{\sigma_X \sigma_Y} \quad (1.5)$$

where E is the expectation operator.⁹

Definition 1.10 (statistics). *Autocorrelation* $G(X; \tau)$, is the correlation of a sig-

⁸https://en.wikipedia.org/wiki/Fluorescence_correlation_spectroscopy

⁹https://en.wikipedia.org/wiki/Correlation_and_dependence

nal X with a delayed copy of itself as a function of the delay τ .¹⁰

$$G(X; \tau) = \text{corr}(X_t, X_{t+\tau}) = \frac{E[(X_t - \mu_X)(X_{t+\tau} - \mu_X)]}{\sigma_X^2} \quad (1.6)$$

Definition 1.11 (statistics). The *cross-correlation* of two series is the correlation of one with a delayed copy of the other as a function of the delay.

$$\text{crosscorr}(X, Y; \tau) = \text{corr}(X_t, Y_{t+\tau}) = \frac{E[(X_t - \mu_X)(Y_{t+\tau} - \mu_Y)]}{\sigma_X \sigma_Y} \quad (1.7)$$

For the purposes of FCS, these quantities were redefined as follows.

Definition 1.12 (FCS). The *correlation* between two random variables X and Y with expected values μ_X and μ_Y and standard deviations σ_X and σ_Y is

$$\text{corr}(X, Y) = \frac{E[(X - \mu_X)(Y - \mu_Y)]}{\mu_X \mu_Y} \quad (1.8)$$

where E is the expectation operator.¹¹

Definition 1.13 (FCS). *Autocorrelation* $G(X; \tau)$, is the correlation of a signal X with a delayed copy of itself as a function of the delay τ .¹²

¹⁰<https://en.wikipedia.org/wiki/Autocorrelation>

¹¹https://en.wikipedia.org/wiki/Correlation_and_dependence

¹²<https://en.wikipedia.org/wiki/Autocorrelation>

$$G(X; \tau) = \text{corr}(X_t, X_{t+\tau}) = \frac{E[(X_t - \mu_X)(X_{t+\tau} - \mu_X)]}{\mu_X^2} \quad (1.9)$$

Definition 1.14 (FCS). The *cross-correlation* of two series is the correlation of one with a delayed copy of the other as a function of the delay.

$$\text{crosscorr}(X, Y; \tau) = \text{corr}(X_t, Y_{t+\tau}) = \frac{E[(X_t - \mu_X)(Y_{t+\tau} - \mu_Y)]}{\mu_X \mu_Y} \quad (1.10)$$

The reason for these redefinitions (which just involve replacing standard deviations with means in the denominators of each expression) is that with the FCS definition, the autocorrelation has the nice property that for normal diffusion

$$G(X; 0) = \frac{1}{n} \quad (1.11)$$

where n is the mean number of fluorescent particles in the focal volume. The convenience of the statistics definitions is that there, correlations are guaranteed to be in $[-1, 1]$, with 0 representing no correlation, 1 perfect positive correlation and -1 perfect negative correlation; this is lost with the FCS definitions: 0 still represents no correlation, but correlation values are no longer bounded, so the ideas of perfect correlation are lost.

I felt it necessary to provide these definitions for two reasons:

1. It is important for people from the fields of FCS and pure mathematics/statistics to know that they have different definitions for the same thing.
2. In FCS, it's very common for people to mistake correlation for cross-correlation. This is unfortunate, but knowing about this common mistake is essential for navigating the field in a sensible manner. It seems that when people in the FCS field correlate the signals from two separate channels, they use the term *cross*-correlation, even though they're only using correlation. I think the idea of working *across* two or more channels (and ideas such as *cross*-talk) leads to this confusion.

Remark. Henceforth, the FCS definitions of these quantities will be assumed.

1.4.1 Correlation

Suppose that two proteins of interest A and B are labelled with red and green fluorophores respectively (and there is no bleed-through between these red and green channels).

If these proteins are interacting, then

- interaction implies that A and B are *stuck* together

- this implies that A and B co-diffuse
- this implies that for a given volumen in the sample
- more of A implies more of B
- less of A implies less of B
- more of B implies more of A
- less of B implies less of A

Since the number of photons emitted is proportional to the amount of fluorophores present, it follows that

- more red photons implies more green photons
- less red photons implies less green photons
- more green photons implies more red photons
- less green photons implies less red photons

Altogether, this implies that the intensity traces from the red and green channels will be correlated. If these proteins are not interacting, then the intensity traces from the red and green channels will not be correlated. Thus, interaction of red A and green B necessarily leads to correlation in fluorescent signals from the red and green channels.¹³

¹³This correlation may not always be detectable due to weak interaction or weak signal.

1.4.2 Autocorrelation

As mentioned already, the autocorrelation function (ACF) can be used to count the number of particles in the confocal volume. It can also be used to measure diffusion coefficients for various types of diffusion (normal, anomalous, polydisperse, etc.). The ACF is not used in my PhD.

1.4.3 Cross-correlation

The cross-correlation of intensity traces from nearby pixels can be used to measure the velocity of the movement of the labelled particles between these two pixels (Hebert et al., 2005). It can also be used to image barriers to diffusion (Digman and Gratton, 2009).

1.4.4 Cross-correlated brightness

Cross-correlated brightness (Digman et al., 2009) moulds the correlation idea (section 1.4.1) into the framework of N&B. Suppose there are two channels with intensities I_1 and I_2 .

Definition 1.15 (cross-variance).

$$\sigma_{\text{cc}}^2 = E[(I_2 - \langle I_1 \rangle)(I_2 - \langle I_2 \rangle)] \quad (1.12)$$

Definition 1.16 (cross-correlated brightness).

$$B_{\text{cc}} = \frac{\sigma_{\text{cc}}^2}{\sqrt{\langle I_1 \rangle \langle I_2 \rangle}} \quad (1.13)$$

B_{cc} is related to $\text{corr}(I_1, I_2)$ by

$$B_{\text{cc}} = \sqrt{\langle I_1 \rangle \langle I_2 \rangle} \times \text{corr}(I_1, I_2) \quad (1.14)$$

This means that B_{cc} is just a scaled version of correlation. The need for this redefinition is unclear (but it is no harm). It does make the formula look like the brightness formula (1.2), but no such oligomeric state information can be gleaned from B_{cc} . It is merely useful as a relative measure of interaction: higher B_{cc} means more interaction, lower B_{cc} means less interaction. It is commonly used to identify interactions. Then, conventional N&B performed on each of the channels (1 and 2) can be used to measure the stoichiometry of the interaction. See [Digman et al. \(2009\)](#) for details.

Remark. Since cross-correlated brightness uses correlation but not cross-correlation, it is a prime example of the confusing naming that pervades FCS. It should be called *correlated brightness*. Rather than rename it, I will continue to refer to it as cross-correlated brightness.

Chapter 2

Instrumentation and Software

2.1 Instrumentation

All of the images used in this thesis were acquired on a Leica SP8 confocal microscope.¹ Importantly, the detectors that we have on this microscope are capable of photon counting.

These images were acquired by Luis Alvarez (postdoctoral researcher), Maro Iliopoulou (research assistant) and Sergi Padilla-Parra (supervisor).

¹<https://www.leica-microsystems.com/products/confocal-microscopes/details/product/leica-tcs-sp8>

2.2 Software programs, languages and tools

2.2.1 C++

C++ ([Stroustrup, 2013](#)) is a general-purpose programming language optimized for performance (speed), efficiency (with use of computer resources) and flexibility.²

I used it for its speed, since many of the algorithms that I developed were quite computationally intensive and hence speed was an important consideration.

2.2.2 R

R ([R Core Team, 2016](#)) is a programming language and free software environment for statistical computing and graphics.³ I use R primarily as a wrapper for my C++ code to make my algorithms more user-friendly. R is best used with the *RStudio* integrated development environment.⁴

²<https://en.wikipedia.org/wiki/C%2B%2B>

³[https://en.wikipedia.org/wiki/R_\(programming_language\)](https://en.wikipedia.org/wiki/R_(programming_language))

⁴<https://www.rstudio.com>

2.2.3 ImageJ

ImageJ [ImageJ](https://imagej.net) is an open source image processing program designed for scientific multidimensional images.⁵ It is the preferred image viewing and analysis software in the community. I have written my software in C++ and R because they are easier for developers, but I still intend to translate my image-related algorithms ImageJ plugins. ImageJ is best used via the FIJI ([Schindelin et al., 2012](#)) distribution.⁶

2.2.4 Git and GitHub

Git is a free and open source distributed version control system designed to handle everything from small to very large projects with speed and efficiency.⁷ GitHub is a web-based hosting service for version control using Git.⁸ All of the computer code used during my thesis can be found on my GitHub at <https://github.com/rorynolan>. The vast majority of my time during my thesis was spent writing code so this GitHub account is the best record of the work that I have done.

⁵<https://imagej.net>

⁶<https://imagej.net/Fiji>

⁷<https://git-scm.com/>

⁸<https://en.wikipedia.org/wiki/GitHub>

2.3 My software packages

2.3.1 `ijtiff`

An R package for general purpose TIFF file I/O. This is currently the only such package with read and write support for TIFF files with floating point (real-numbered) pixels, and the only package that can correctly import TIFF files that were saved from *ImageJ* (Rueden et al., 2017). R has millions of users worldwide so this TIFF I/O capability is a basic need for masses of people. `ijtiff` gets 363 downloads per month which amounts to 2299 since it was first released.

This package is part of the *rOpenSci* project. *rOpenSci* is a non-profit initiative founded to make scientific data retrieval reproducible.⁹ This package was peer reviewed and published (Nolan and Padilla-Parra, 2018).

2.3.2 `autothresholdr`

`autothresholdr` provides the *ImageJ* (Rueden et al., 2017) *Auto Threshold* plugin (Landini et al., 2016) functionality to R users. It gets 560 downloads per month which amounts to 9034 since it was first released.

⁹<https://ropensci.org/about>

2.3.3 **detrendr**

detrendr is an R package for detrending images (correcting for photobleaching). It provides all detrending algorithms mentioned in section 3. The detrending is done in C++ in the background for speed but it is wrapped in an R package for ease of use. It gets 394 downloads per month which amounts to 3225 since it was first released.

2.3.4 **filesstrings**

filesstrings is an R package providing convenient functions for moving files, deleting directories, and a variety of string operations that facilitate manipulating file names and extracting information from strings. The motivation for making this package was to facilitate the use of file names for metadata. This is very common in microscopy, e.g. a filename like `well1_cell1_before_drug_addition.tif` is often seen. Using file names for metadata like this is a good idea, however if the naming or the extraction of data from these names is inconsistent, analysis becomes a nightmare and less reproducible. This package was peer reviewed and published (Nolan and Padilla-Parra, 2017b). It gets 755 downloads per month which amounts to 12197 since it was first released.

2.3.5 `exampletestr`

Definition 2.1. *Unit testing* is a software testing method by which individual units of source code, sets of one or more computer program modules together with associated control data, usage procedures, and operating procedures, are tested to determine whether they are fit for use.¹⁰

Unit testing is a tool to verify that software is performing as intended. It is a great way to discover bugs in software. `exampletestr` is an R package which makes it easier for R package developers to write unit tests for their packages. It helped me to eradicate many bugs in all of my packages. Interestingly, `exampletestr` was used to unit test and eradicate bugs in itself! This package was peer reviewed and published in 2017.(Nolan and Padilla-Parra, 2017a) It gets 456 downloads per month which amounts to 7023 since it was first released.

¹⁰https://en.wikipedia.org/wiki/Unit_testing

Chapter 3

Photobleaching Correction

3.1 Introduction to photobleaching

In the ideal case, an *incident* photon of appropriate wavelength is absorbed by a fluorophore, promoting the fluorophore to an excited state; subsequently, the fluorophore relaxes down to its ground state by emitting a photon. In reality, it is possible that the incident photon can *break* the fluorophore with the result that it will no longer emit light. This breaking is referred to as *photobleaching* (or *bleaching* for short). Bleaching causes a diminution in the number of effective fluorophores which is a direct cause of a loss of fluorescent signal.

Many quantitative methods in fields such as fluorescence fluctuation spectroscopy (FFS) and fluorescence correlation spectroscopy (FCS) implicitly assume that there is no bleaching in the data. Hence, data (image series) with significant levels of photobleaching must be corrected prior to the application of equations and algorithms in these fields. A main focus of this thesis is on how to correct fluorescent image series for the effects of bleaching, given that bleaching *does* occur. There is no attempt to understand *why* and/or *how* photobleaching occurs.

Remark. All of the current literature mentions bleaching correction as being purely for correcting the problem of non-stationary mean (NSM), neglecting the problem of non-stationary variance (NSV). Figure 3.3 shows that NSM and NSV go hand-in-hand. Correction for NSM is referred to as trend removal or *detrending*. Hence, the terms *detrending* and *bleaching correction* have come to be used interchangeably. I will follow this convention and use the term *detrending* from now on to mean correction for NSM and possibly also NSV. Starting at section 3.3, the focus is on correcting for NSM. Discussion of correction for NSV starts in section 3.7.

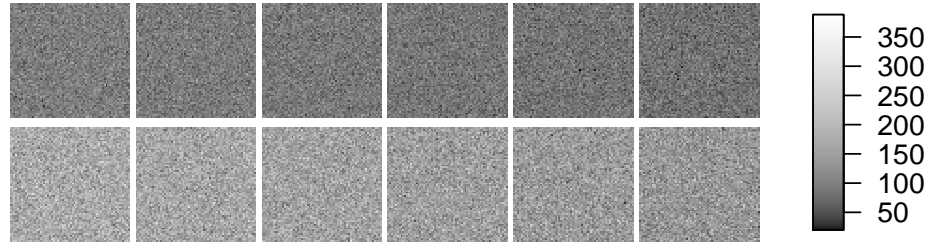


Figure 3.1: Frames 1, 100, 200, 300, 400 and 500 from image series of 100,000 diffusing emitters of brightness $\epsilon = 4$ (top) and $\epsilon = 7$ (bottom) with 15% (top) and 20% (bottom) bleaching.

3.2 The effects of bleaching in FCS and FFS

We simulate two image series, each with 100,000 diffusing fluorescent particles. In the first image series (`img1`), these have brightness $\epsilon = 4$ and in the second (`img2`) they have brightness $\epsilon = 7$. See figure 3.1. We bleach these by 15% and 20% to create `img1_bleached` and `img2_bleached` respectively.

It may not be obvious that these image series are subject to bleaching from figure 3.1, but we can see it more clearly in figure 3.2.

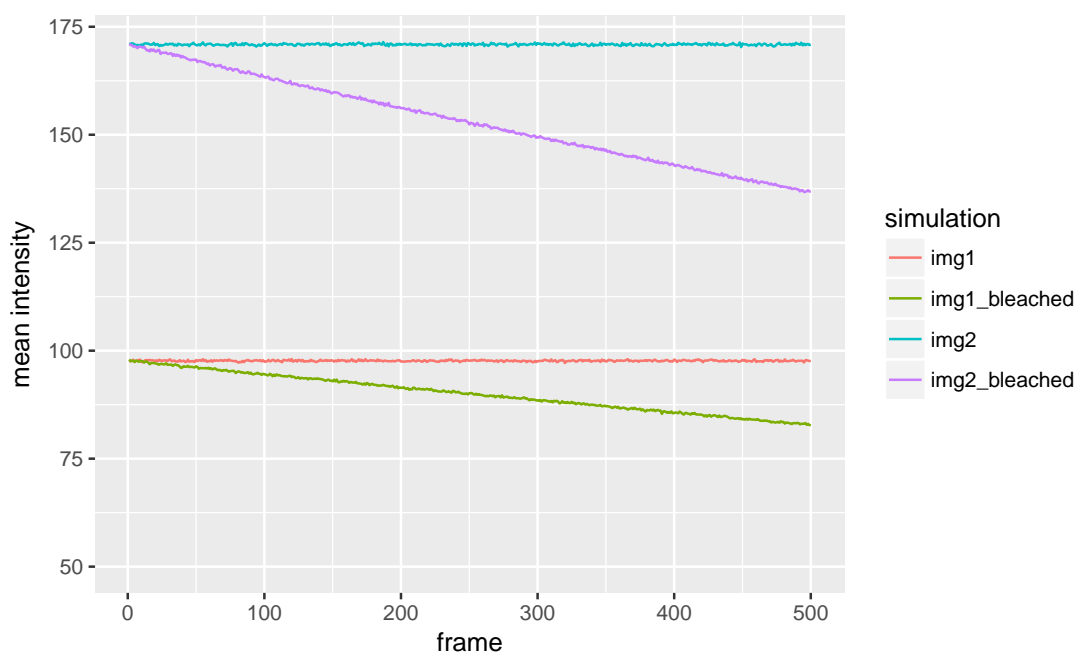


Figure 3.2: Mean intensity profiles of the simulated image series with and without bleaching. Simulation 1 was bleached by 15% and simulation 2 by 20%.

3.2.1 FCS

The unrelated images `img1` and `img2` have a tiny median cross-correlated brightness of $B_{cc} = 0.0036$, signifying no significant correlation, as one would expect. However, the bleached images `img1_bleached` and `img2_bleached` have a significant $B_{cc} = 0.3686$. This shows that bleaching is introducing correlation between otherwise unrelated images. Since correlation is used as a proxy for hetero-interaction, bleaching can make it appear as though there is interaction when in fact there is not.

3.2.2 FFS

Definition 3.1 (median mean pixel intensity). The *median mean pixel intensity* of an image series is found by taking the mean intensity of each pixel in the image series and then taking the median of those means. It can be thought of as a summary statistic for the pixel intensity of the image series.

Definition 3.2 (median pixel intensity variance). The *median pixel intensity variance* of an image series is found by taking the variance in the intensity of each pixel in the image series and then taking the median of those variances. It can be thought of as a summary statistic for the variance in the pixel intensity of the image series.

In FFS, one is always interested in the mean and variance of pixel values (and sometimes other fluctuation quantities like the third moment). `img1` has a median mean pixel intensity of 98 and a median pixel intensity variance of 487. The mean brightness is $\epsilon = 3.9959$ (very close to 4, as expected since the image series was simulated with brightness $\epsilon = 4$). For `img1_bleached`, we find a median mean pixel intensity of 90 and a median pixel intensity variance of 468. The mean brightness is $\epsilon = 4.2026$. Hence, bleaching has altered both the means and variances of the pixels, resulting in a change in calculated brightness. The non-stationary mean frame intensity introduced by bleaching decreases the mean but increases the variance. The loss in signal has the effect of slightly decreasing the variance: with poisson statistics (such as photon-emission), a loss of signal (photons) leads directly to a loss in variance. This is a subtle point not discussed anywhere in the literature; it is shown in figure 3.3.

3.3 Exponential fitting detrending

Naively, one could assume that bleaching of fluorophores takes place at some constant rate. This would mean that the intensity of the image would die-off according to an exponential decay. In figure 3.4, we fit an exponential decay to such ideal data.

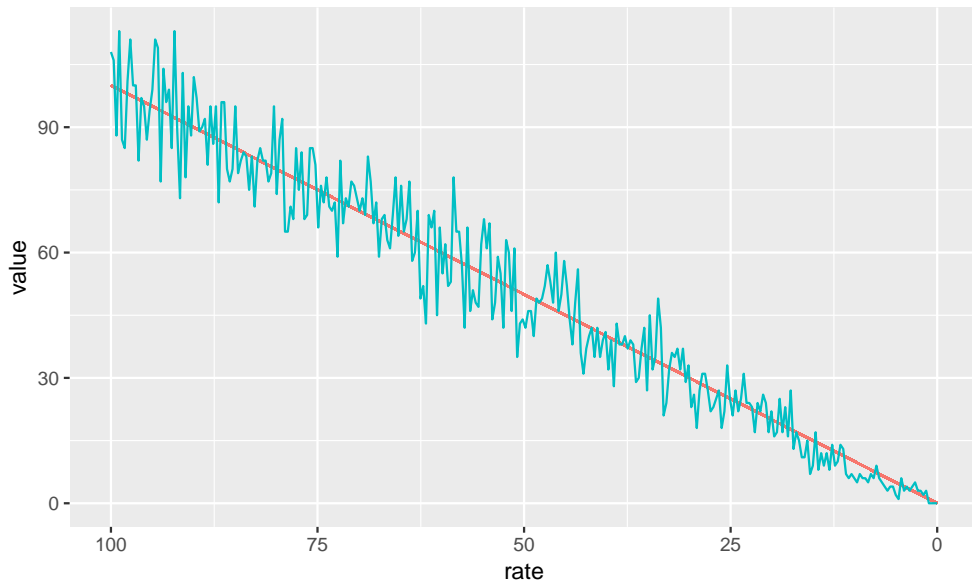


Figure 3.3: A decrease in the poisson rate (e.g. for emmission of photons) leads to a decrease in the mean (blue line) but also a decrease in fluctuations around the mean. Notice that towards the right where rate is low, fluctuations around the mean are at their smallest.

Having fit the data, one may record the deviations from the fitted line as the *fluctuations* and replace these fluctuations about a straight line, which is placed at the mean of the original series; for figure 3.4 above, this mean is 90. Figure 3.5 shows the corrected series.

We can see that here, in the ideal case where the naive assumptions of the exponential decay fitting approach hold, this approach works quite well. Let us now examine the case where these assumptions don't hold because there are other long-term fluctuations e.g. due to cell movement. We add these other fluctuations as a gentle sinusiod. See figure 3.6.

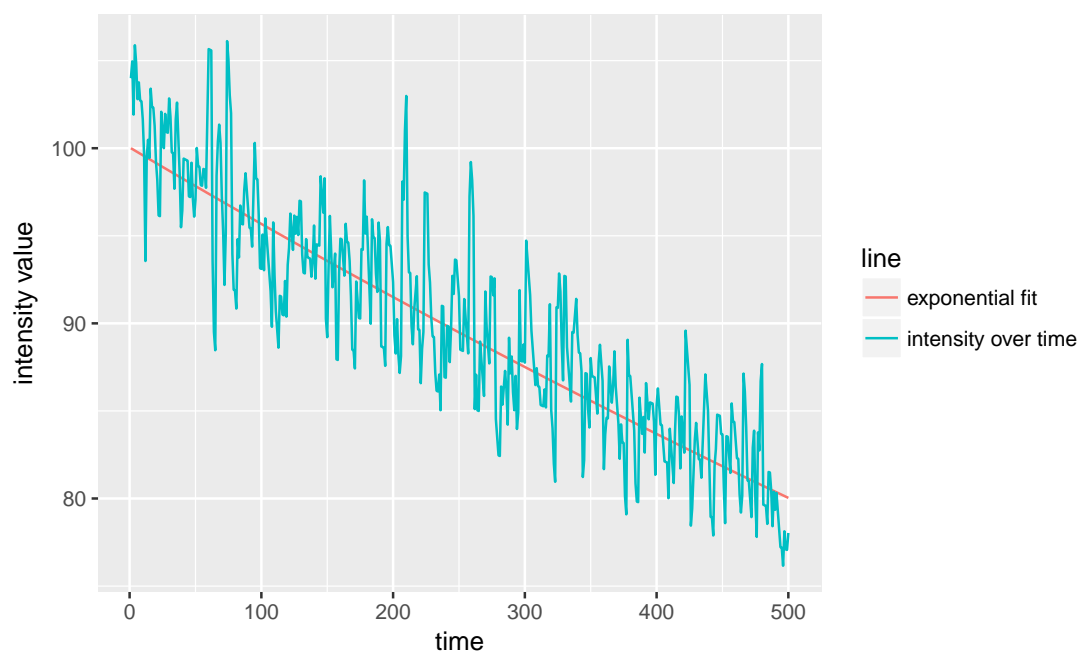


Figure 3.4: Exponential fit to intensity trace of image which is subject to bleaching at a constant rate.

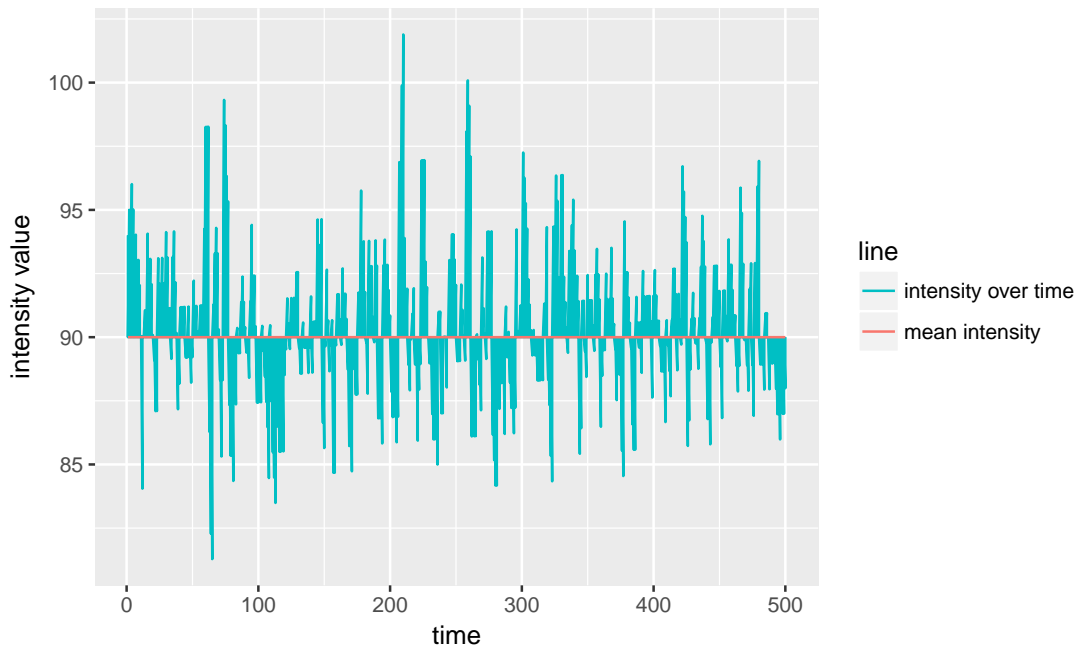


Figure 3.5: The blue line from figure 3.4 has changed to a straight horizontal line cutting the y axis at the mean intensity of the original intensity trace. The fluctuations about the blue line that existed in figure 3.4 are preserved here. As an example, see the large downward fluctuation at $t \approx 50$ seconds in both figures.

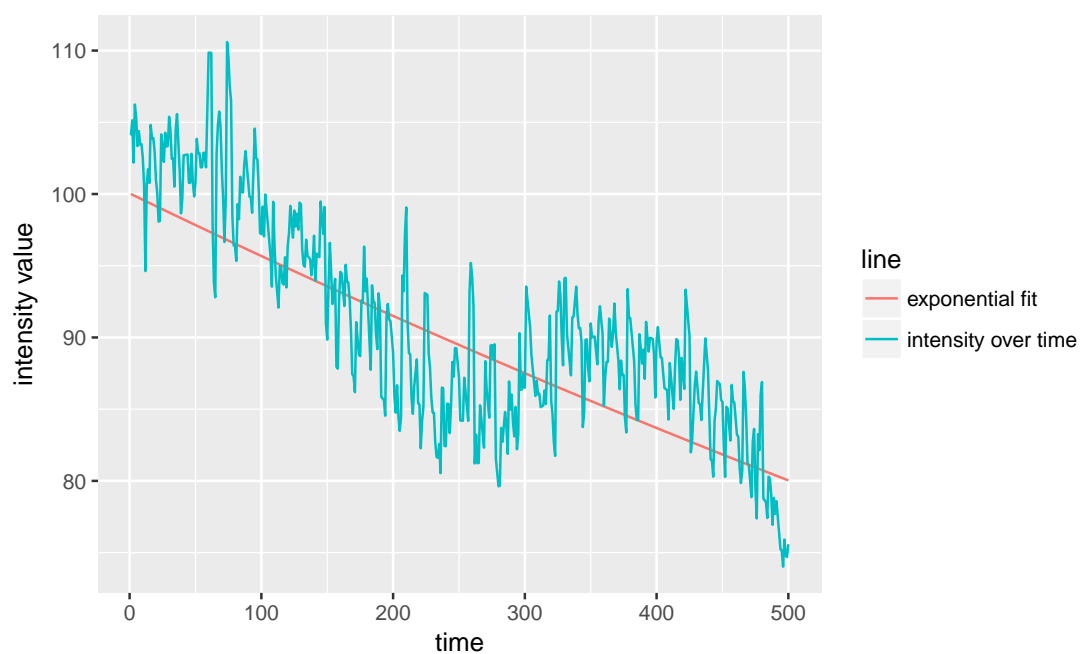


Figure 3.6: An exponential decay with added sinusoidal variance, fit with a simple exponential decay.

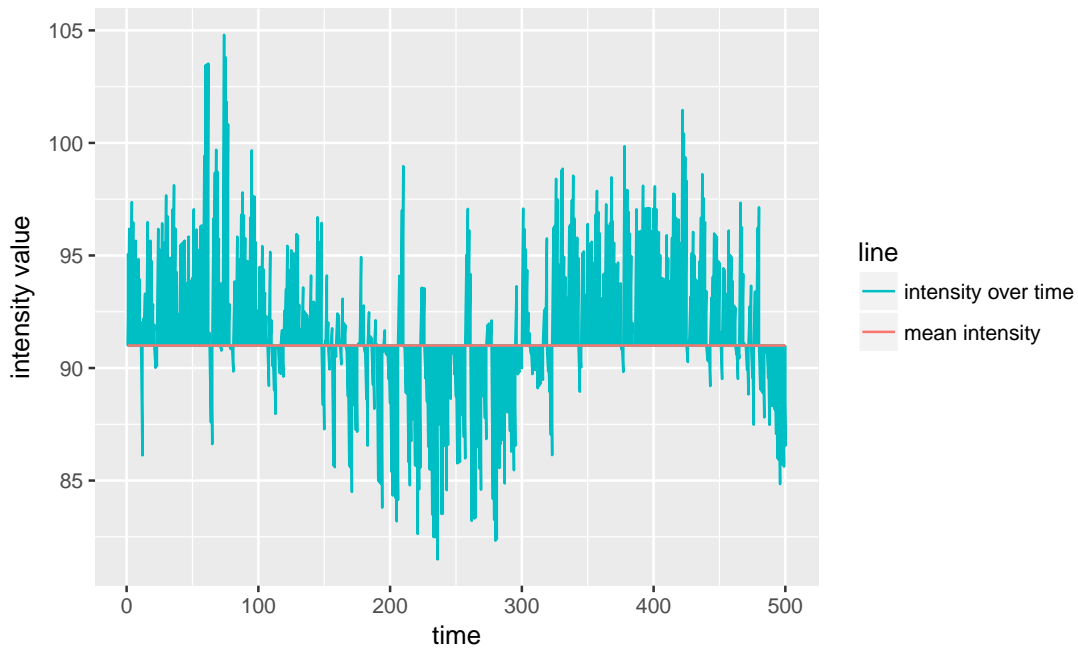


Figure 3.7: Result of exponential fitting detrending applied to a decay with a long-term sinusoidal trend component.

One can see by eye that this is not a good fit for the data. This has disastrous consequences for the detrended series, shown in figure 3.7.

One can see in figure 3.7 that the exponential fit detrend failed to remove the sinusoidal trend in the data (even though it *did* remove the exponential decay component). We have now seen that exponential fitting detrending is appropriate when the decay has a particular form, but is otherwise not fit for use. This is a problem common to all fitting approaches to detrending, even the more flexible types like polynomial detrending (Chan et al., 1977). For this reason alone, for the purpose of detrending, fitting approaches should be avoided.

3.4 Boxcar smoothing detrending

A common approach to obtaining the line from which to measure deviations/fluctuations (as in the red line in figure 3.4) is to *smooth* the time series, i.e. construct the line by taking a *local average* at each point. This is often referred to as *boxcar smoothing* because it can be visualised as drawing a box around a neighbourhood of points, taking their average as the smoothed value at that point and then moving the box onto the next series of points and repeating the procedure. See figure 3.8.

```
#> Warning: Removed 9 rows containing missing values (geom_rect).
```

```
#> Warning: Removed 9 rows containing missing values (geom_rect).
```

The parameter l of a boxcar is such that the length of the boxcar is equal to $2l + 1$. This ensures that the length of the boxcar is always odd) which means it can always be centred upon a point). Hence the allowable lengths of a boxcar are 3, 5, 7, 9, etc.

The boxcar parameter l has a large effect on the type of smoothing achieved. This can be seen in figure 3.9 where boxcar smoothing is applied to the trace in figure 3.4. The traces for $l = 1$ and $l = 3$ are far too wiggly (not smooth enough); the

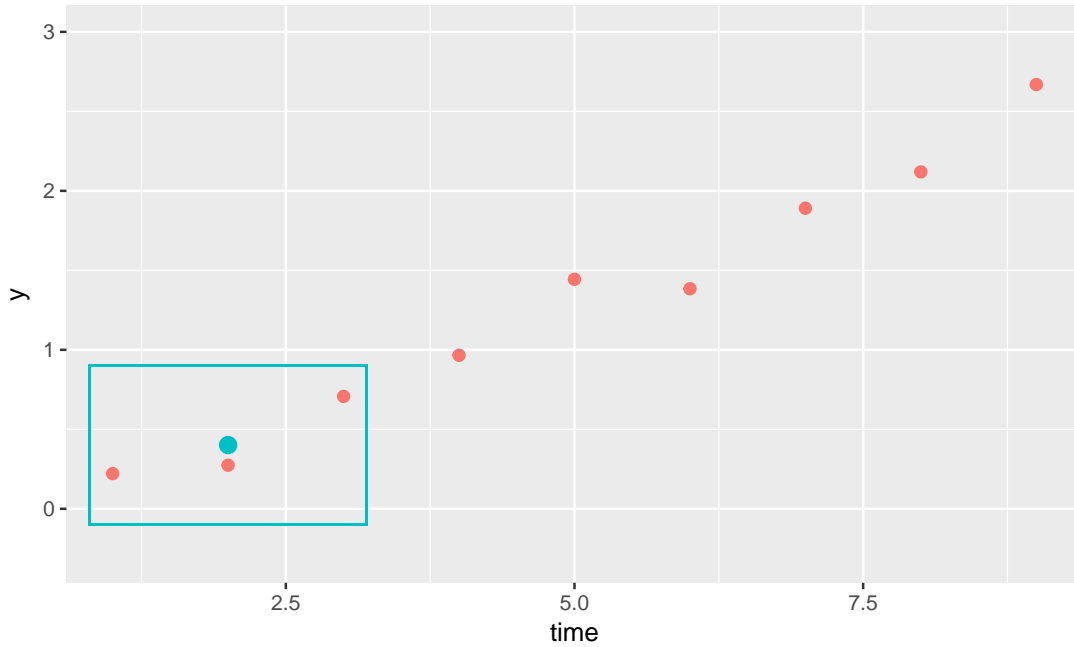


Figure 3.8: The original time series is depicted by the red dots. The blue rectangles represent the *boxcar*. This boxcar is said to be of length 3 because it is wide enough to encompass 3 points at a time. The boxcar is centred on a point and then the smoothed value at that point (blue dot) is calculated as the mean value of all points within the boxcar. In reality, every point gets a smoothed value which means that the boxcar *overlaps* but in this figure, for the sake of clarity, they are not overlapped.

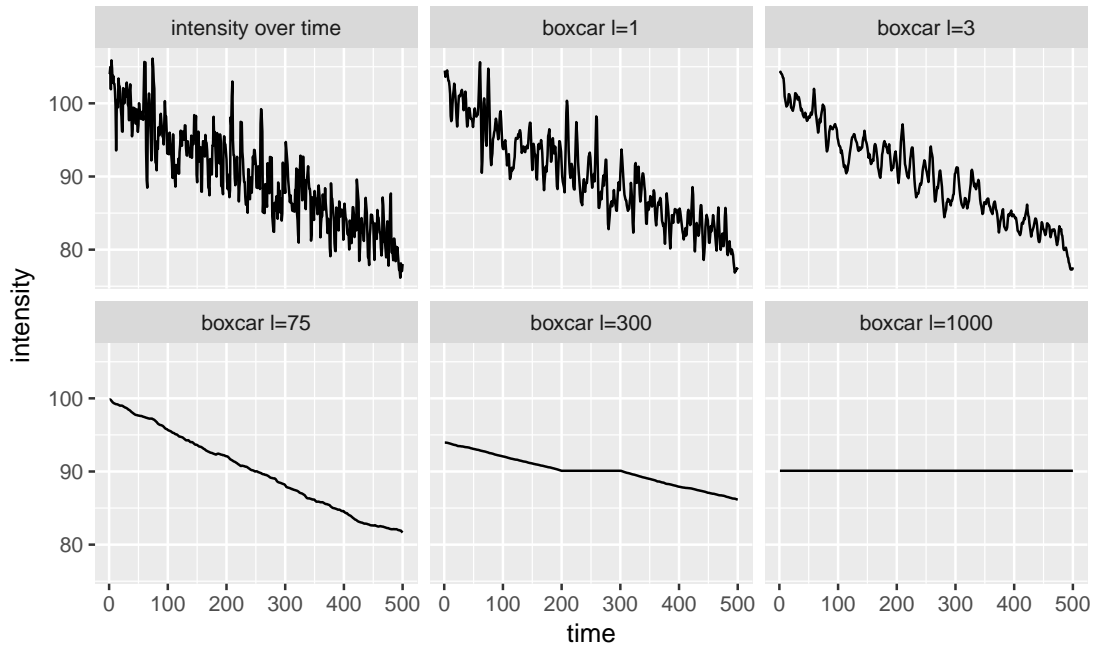


Figure 3.9: The original intensity trace is shown in the top-left. The other panels show the result of boxcar smoothing for $l = 1, 3, 75, 300$ and 1000 .

trace for $l = 75$ is better but perhaps still slightly wiggly; finally, the traces for $l = 300$ and $l = 1000$ are too close to straight horizontal lines (too smooth).

This begs the question: what is the correct smoothing parameter l ?

3.5 Choosing the correct smoothing parameter for detrending

Figure 3.9 shows that the choice of boxcar size is crucial because different sizes lead to very different “smoothed” lines. The most common choice in the community is to choose $l = 10$ (Laboratory for Fluorescence Dynamics, 2018). There is no justification for this choice.

In section 1.3.1, we learned that for immobile particles, the expected brightness is $B = 1$. This fact can be used to solve for the appropriate choice of l to use for detrending a specific image series.

Definition 3.3. The *mean intensity profile* of one channel of an image series is obtained by calculating the mean intensity of each frame in that image series.

The mean intensity profile can be used to visualise the bleaching of an image series. If the fluorophores are bleaching, the mean intensity should be decreasing over time. To proceed with solving for the appropriate l , we need to make one assumption. This is that any two image series with the same mean intensity profile are appropriately detrended with the same detrending parameter l . This assumption seems reasonable to me, however there is no need to debate its validity because later, detrending with the solved-for parameter l will be evaluated with

simulated data and compared to the standard $l = 10$. If this assumption is bad, then the performance of the detrending that relies on it should also be bad. With this assumption in hand, solving for l proceeds as follows:

1. Simulate an image series with immobile particles only which has the same mean intensity profile as the acquired real data.
2. Given that the simulated series is of immobile particles only, once properly detrended, it should have $B = 1$.
3. The l for which the detrended series has mean brightness closes to 1 is the most appropriate for the simulated data.
4. By the assumption above, this l is the most appropriate for the real data.

Mathematically, this can be expressed as

$$l = \operatorname{argmin}_l (1 - |\text{mean brightness of simulated series detrended with parameter } \tilde{l}|) \quad (3.1)$$

In fact, what I have done here is to give a general method for solving for any detrending parameter α :

$$\alpha = \operatorname{argmin}_{\tilde{\alpha}} (1 - |\text{mean brightness of simulated series detrended with parameter } \tilde{\alpha}|) \quad (3.2)$$

This will be useful later when other detrending regimes with their own parameters are introduced.

3.6 Exponential smoothing detrending

Exponential smoothing is a slight alteration to boxcar smoothing. The idea is that when computing a local average, points nearer to the point of interest should have greater weights.¹ The weights fall off with distance $|t|$ from the point of interest according to $\exp(-\frac{|t|}{\tau})$ where the parameter τ is a positive real number. This function is visualized in figure 3.10. For small values of τ , only values very close to the point of interest have importance when calculating the local average. For larger values of τ , further values also have importance (but closer values always have higher weights). In this sense, increasing the value of τ has a similar effect to increasing the value of l for the boxcar in that further away points are taken into account.

¹https://en.wikipedia.org/wiki/Weighted_arithmetic_mean

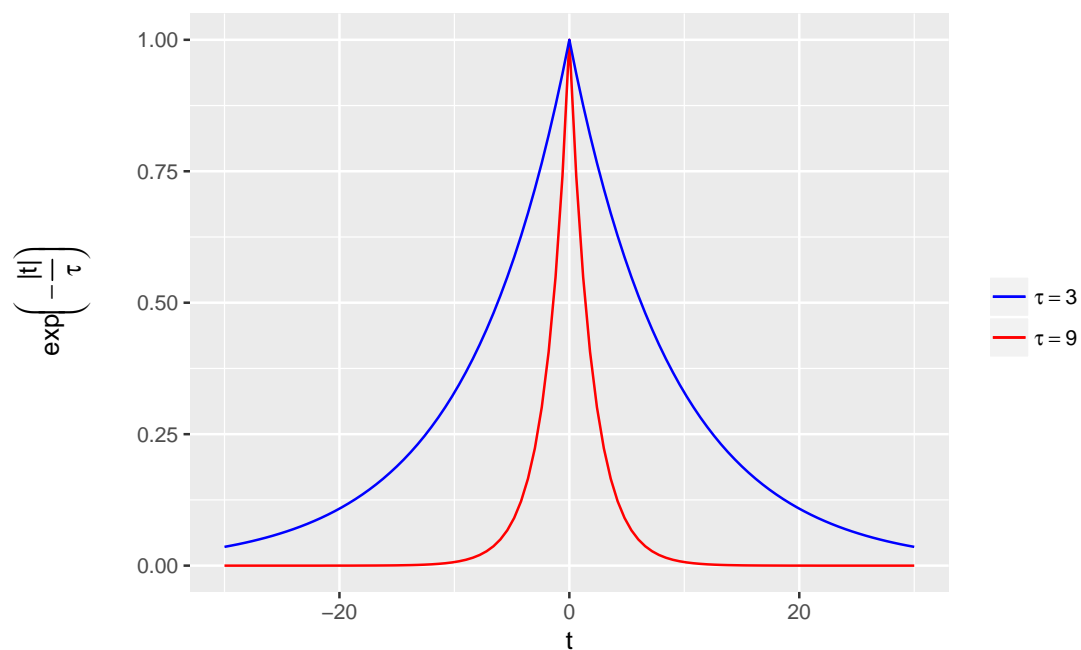


Figure 3.10: The function $\exp(-\frac{|t|}{\tau})$ visualised with $\tau = 3$ and $\tau = 9$. For $\tau = 3$, points at distance $|t| = 10$ have approximately zero weight, whereas for $\tau = 9$, these points have significant weight.

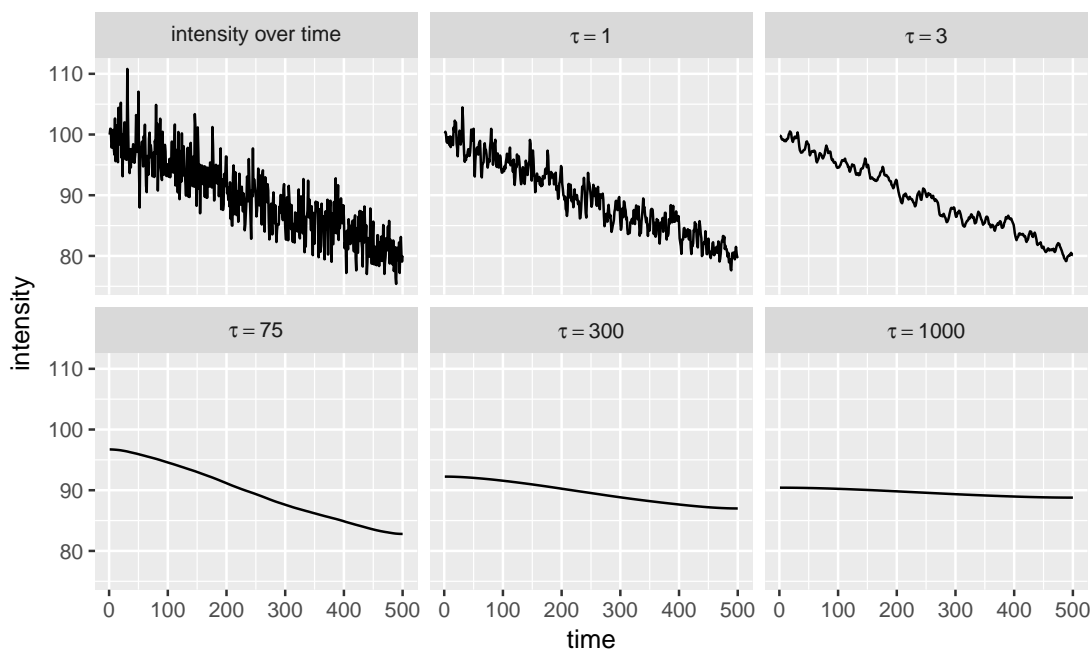


Figure 3.11: The original intensity trace is shown in the top-left. The other panels show the result of exponential smoothing for $\tau = 1, 3, 75, 300$ and 1000 .

In figure 3.11, exponential smoothing with different parameters τ is applied to the trace in figure 3.4. The results are similar to those in figure 3.9.

Heuristically, exponential smoothing detrending seems favourable to boxcar detrending because the idea that points further away from the point of interest are less important (but still somewhat important) when computing the local average is reasonable. Indeed, this was the method proposed in the original number and brightness paper (Digman et al., 2008). For this reason, exponential smoothing was the method of choice for my paper where the method of choosing the correct detrending parameter was published (Nolan et al., 2017a).

3.7 Correcting for non-stationary variance

Definition 3.4. The variance of a random variable X is the expected value of the squared deviation of X from its mean μ :

$$\text{Var}(X) = E[(X - \mu)^2] \quad (3.3)$$

All of this chapter so far has focused on correcting for non-stationary mean. As shown in figure 3.3, as the mean decreases, so too does the variance. For an instance x of the random variable X with expected value $E[X] = \mu$, $x - \mu$ is the *deviation* of x from μ . If we write x as $x = \mu + \tilde{x}$, then we get the deviation $x - \mu = (\mu + \tilde{x}) - \mu = \tilde{x}$, so \tilde{x} is the deviation. For a given point in figure 3.3, its deviation is its distance from the red line. For positive real number k , making the transformation $\tilde{x} \rightarrow \sqrt{k}\tilde{x}$ i.e. $x \rightarrow \mu + \sqrt{k}\tilde{x}$ causes the variance (i.e. the *squared deviation*) to be transformed as $\text{Var}(X) \rightarrow k \times \text{Var}(X)$. Hence, we have a way to modify the variance of a time series as a whole by modifying the deviation of each time point from the mean. For months, I toyed with this idea as a solution of correcting for non-stationary variance. However, in reality the contribution to the variance in intensity at a given pixel is down to both poisson photon statistics and fluorophore movement. This combination of factors makes it very difficult

to ascertain the amount by which the variance should be altered. I eventually abandoned my efforts to alter the variance like this in favour of the *Robin Hood* detrending algorithm (section 3.9) which includes correction for non-stationary variance as an intrinsic part of its detrending routine.

3.8 Caveats of fitting and smoothing approaches to detrending

Both fitting and smoothing approaches to detrending have serious caveats. Fitting approaches assume that the fluorescence intensity decay has a certain form. Unpredictable issues such as cell movement mean that no particular decay form can be assumed. Smoothing methods do not perform well at the edges of time series that they are applied to. They also require the user to choose a smoothing parameter. The problem of how to best choose this parameter was solved recently (Nolan et al., 2017a), but this method has not been widely adopted. Most importantly, both fitting and smoothing fail when the data cannot be approximated as continuous (fitted and smoothed lines are continuous approximations of data). Fluorescence intensity data at low intensities—where most pixel values are either 0 or 1—are quasi-binary² and hence a continuous approximation does not make

²By *quasi-binary*, I just mean that almost all values are 0 or 1.

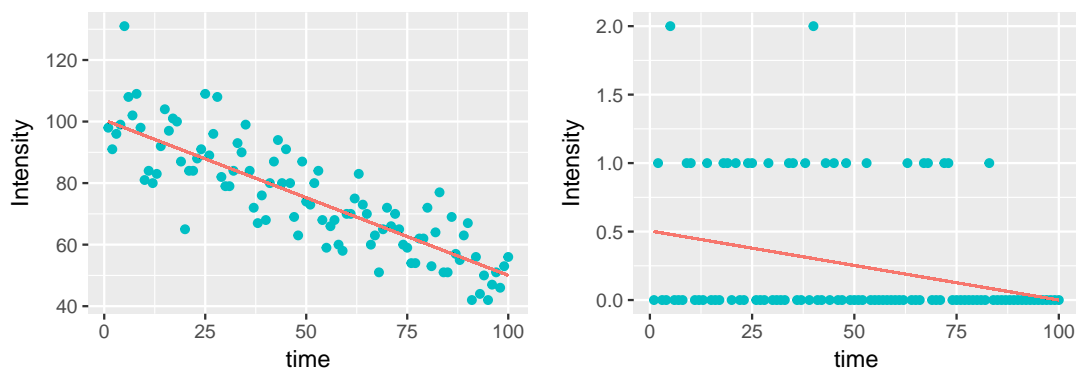


Figure 3.12: Left: for high ($\gg 1$) intensity values, the line is a satisfactory approximation of the data, representing it well. Right: for low (quasi-binary) intensity values, the line is not a good approximation for the data and indeed no line or curve could represent the data well.

sense (see figure 3.12). This means that neither fitting nor smoothing are applicable detrending methods at low intensities. This is the crucial caveat of these methods because, when bleaching is a problem, it is common to reduce laser power to reduce bleaching, which leads directly to lower intensity images. With fitting and smoothing techniques, it may sometimes be advisable to increase the laser power to achieve higher intensities such that the detrending routines will function properly. This means one may need to bleach more in order to be able to correct for bleaching. This farcical situation necessitates a new detrending technique which can function at low intensities.

3.9 Robin Hood detrending

Intensity images in units of photons are count data. This means that the values are all natural numbers, i.e. elements of $\mathbb{N}_0 = \{0, 1, 2, 3, \dots\}$. Fitting and smoothing give real-numbered values (elements of \mathbb{R}), which must then be transformed back into count data (elements of \mathbb{N}_0), normally by rounding. This means that fitting and smoothing methods of detrending push values through $\mathbb{N}_0 \rightarrow \mathbb{R} \rightarrow \mathbb{N}_0$. When current methods were failing to properly detrend low-intensity images, I began to wonder was it necessary to go through the real numbers \mathbb{R} , given that the start and end points were the natural numbers \mathbb{N}_0 ?

Consider figure 3.13. There is a bleached and unbleached version of an intensity trace. Suppose that our real data is the bleached trace, but we *wish* it looked like the unbleached trace. You may wonder why the unbleached trace is not at the starting intensity of the bleached series. For reasons that will become clear, the Robin Hood algorithm can only place the detrended image at the mean intensity of the original image. This is not a problem because the issue with bleaching in FCS and FFS is mainly that the changing signal leads to incorrect calculations, not that the loss in signal leads to a chronic lack of information (photons). Indeed a feature of the Robin Hood algorithm is that it preserves the mean intensity of the real data on a pixel-by pixel basis.

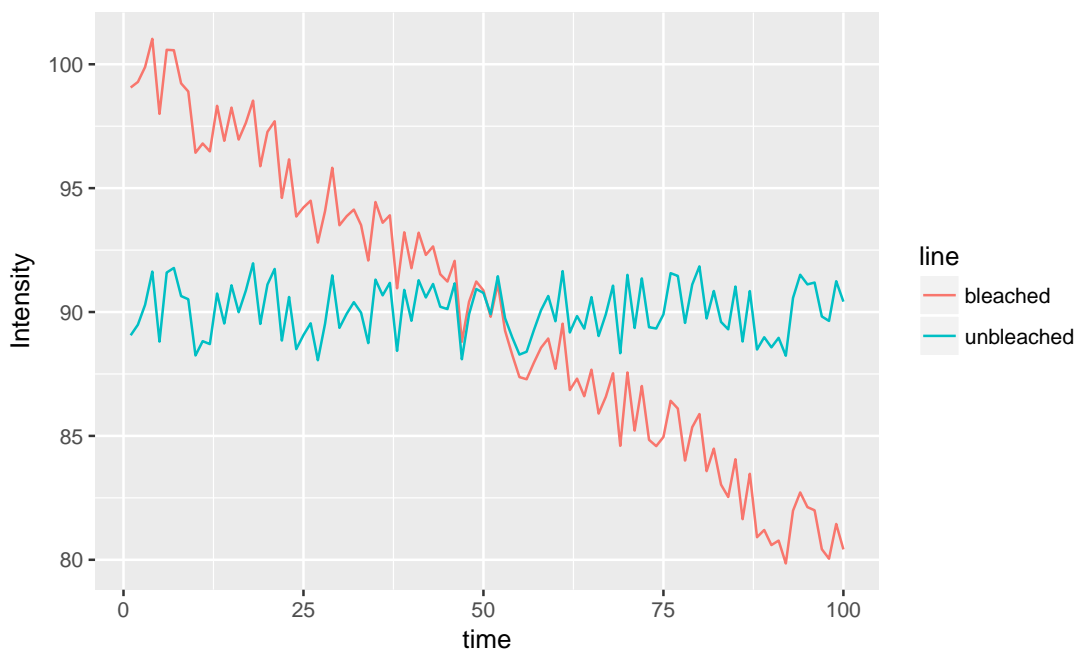


Figure 3.13: Bleached and unbleached intensity traces.

To get to the unbleached intensity trace from the bleached intensity trace, intensity must be subtracted from time-points with too much intensity and added to time points with too little intensity. This can be done by *taking* counts from frames with too much intensity and *giving* them to frames with too little intensity. In this way, no counts are gained or lost, they are just moved around the image series. See figure 3.14. Counts are passed from one frame to another *along* a given pixel, i.e. if a count is taken from pixel at position p in some frame i , it must be given to a pixel at the same position p in some other frame j . It is this condition that ensures that the mean intensity images of the original and detrended image series are the same.

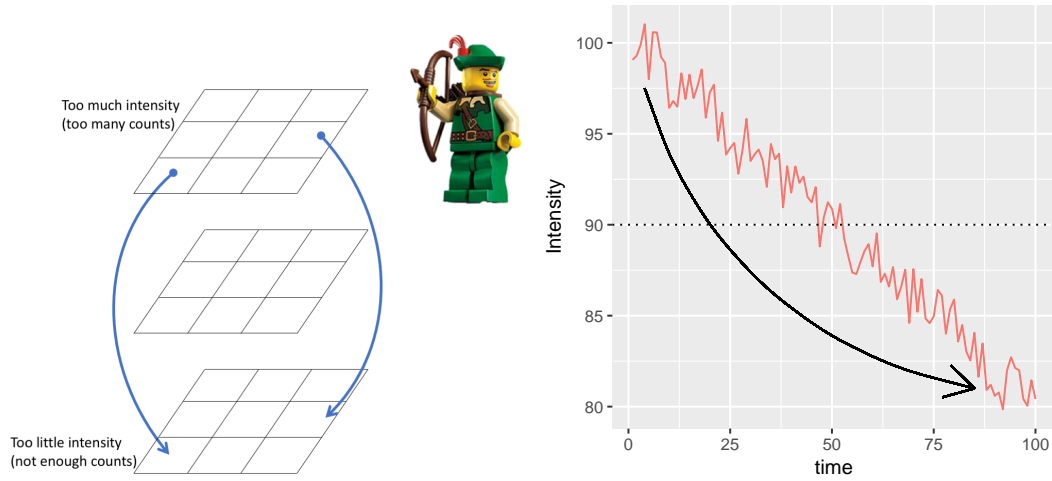


Figure 3.14: Robin Hood: counts are taken from frames of higher intensity (usually closer to the start of the image series) and given to frames of lower intensity (usually closer to the end of the image series).

To determine how many swaps need to be made to detrend a given image series, equation (3.2) can be used, with α being the number of swaps.

The random gifting of counts from higher to lower intensity frames has the effect of redistribution mean intensity but *also* variance in intensity. With photon statistics (which follow a poisson distribution), random counts provide both mean and variance. This is in contrast to all previous methods which consist of determining local deviation and adding it to a *fixed* global mean: this provides no redistribution of variance.

3.10 A comparison of detrending methods

To compare the various detrending methods, I use the following workflow:

1. Simulate a number $N = 100,000$ of particles diffusing with known diffusion rate.
2. Simulate photon emission from these particles with chosen brightness ϵ and create an image series from this, being careful to (virtually) sample at a rate appropriate for number and brightness analysis.
3. Bleach the simulation with a chosen constant bleaching rate.
4. Simulate photon emission from the bleached simulation (bleached particles don't emit photons) with the same brightness ϵ and create an image series.
5. Detrend the bleached image series.
6. Evaluate the detrending algorithm by measuring how close the brightness of the detrended bleached image series is to the known simulated brightness.

For all combinations of brightnesses of $\epsilon = 0.001, 0.01, 0.1, 1, 10$ and bleaching fractions of 0%, 1%, 5%, 10%, 15%, 20%, 25%, 30%, 20 images of 64x64 pixels and 5,000 frames were simulated using 100,000 fluorescent diffusing particles.³

These were detrended with the following detrending routines:⁴

³The simulation took 3 weeks.

⁴The detrending took 2 weeks.

1. Boxcar with $l = 10$ (`boxcar10`, the most common detrending routine).
2. Exponential smoothing with automatically chosen parameter τ (`autotau`).
3. Robin Hood with automatically chosen swaps (`robinhood`).

The performance was evaluated using the *mean relative error*.

Definition 3.5. For a given brightness and bleaching fraction,

$$\text{mean relative error} = \frac{|(\text{calculated brightness after detrending}) - (\text{true brightness})|}{(\text{true brightness})} \quad (3.4)$$

Figure 3.15 shows the results. Before I discuss them, note that the common brightnesses that we see are in the range $\epsilon = 0.003$ to $\epsilon = 0.1$.

The most striking thing about figure 3.15 is that the best choice in all cases is to not detrend at all! This is an interesting result and seems to render all detrending routines worthless. However, when working with real data, not detrending does not work well at all. This will be shown in chapter 4. This is probably because with real data, bleaching is likely not taking place at a constant, steady rate and other factors such as cell movement are contributing to medium and long term intensity fluctuations and these have a detrimental effect on calculations if not detrended out.

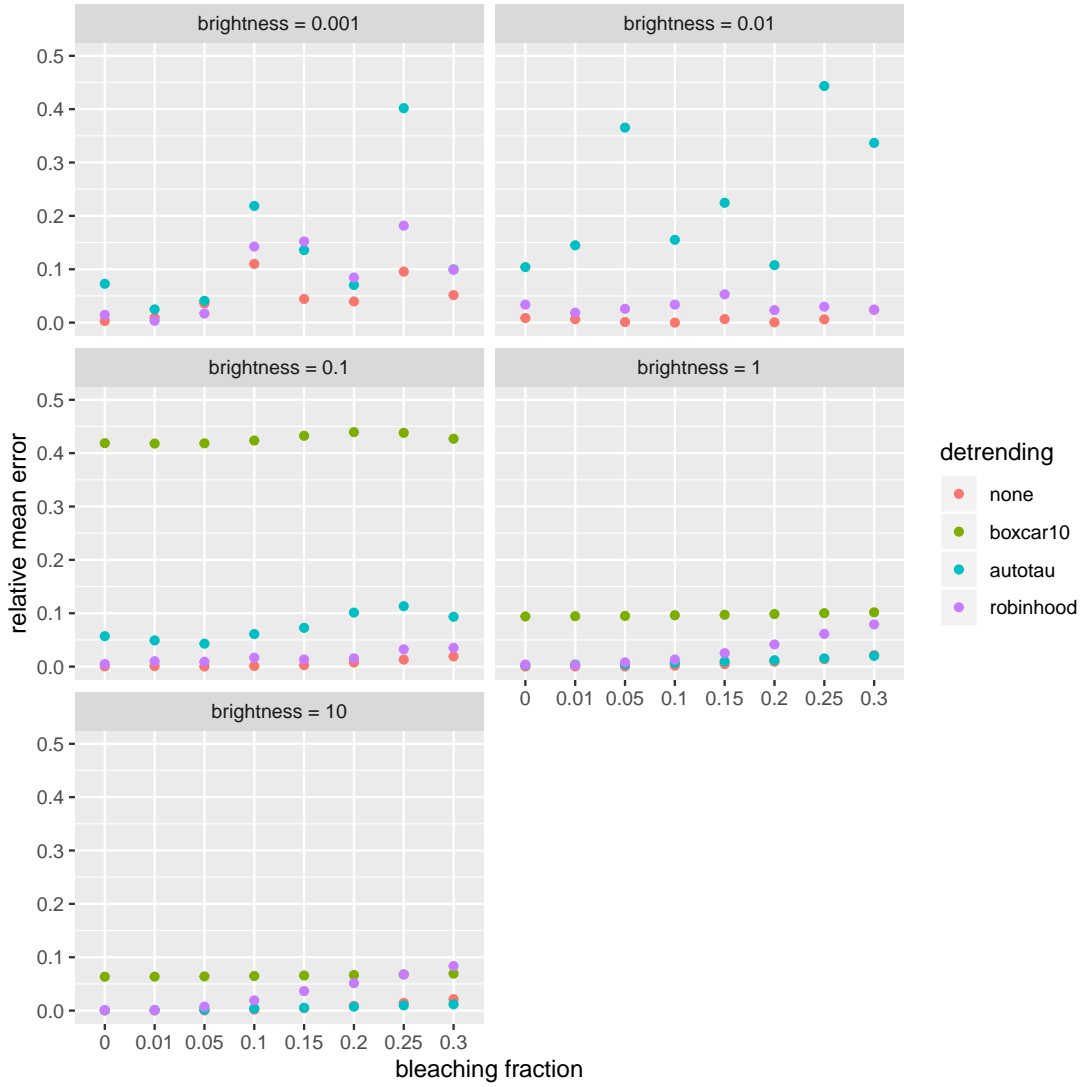


Figure 3.15: A comparison of different detrending methods with various brightnesses and bleaching fractions (steady, constant-rate bleaching), including the results of not detrending at all.

The worst performer by far is `boxcar10`. For example, at $\epsilon = 0.1$, it makes an error of worse than 40% and for $\epsilon = 0.001, 0.01$, its error is worse than 50%, so it does not even appear on the plot. This is good evidence that arbitrarily choosing the parameter l is very bad practice. For realistic brightnesses (≤ 0.1), `robinhood` is the best with errors almost always lower than 5%. `autotau` also performs very well, with errors almost always less than 10%. At the lowest brightness $\epsilon = 0.001$, all methods are somewhat erratic. That is because at this extremely low brightness, there is a critical lack of information (photons) for the algorithms to work with. Finally, at unrealistically high brightnesses of $\epsilon = 1, 10$, `autotau` begins to perform well because at these high photon counts, the caveats of smoothing have totally disappeared. However, I cannot explain the degradation in the performance of `robinhood` in this case. Fortunately, there is no need to dwell on this, as this situation ($\epsilon = 1, 10$) does not arise in practice.

Chapter 4

Applications

4.1 Dimerization of FKBP12

Myristoylated FKBP12 is known to dimerize upon addition of the drug AP1510 (Amara et al., 1997). As a test application of exponential smoothing detrending (section 3.6) with automatically chosen parameter τ , we used this system with number and brightness to verify this dimerization. We tested this in 20 Cos7 cells with mClover-labelled FKBP12. We found a brightness increase in ϵ of ≈ 1.6 -fold using the *automatic* detrending method. The naive choice of $\tau = 10$ resulted in a ≈ 0.7 -fold calculated increase, which is indeed a decrease. The ≈ 1.6 -fold increase suggests that dimerization had occurred, however, we expected the increase to

be ≈ 2 -fold upon dimerization. In that publication (Nolan et al., 2017a), we postulated that the 1.6 figure was due to the fact that not all of the protein had dimerized. Recently, a paper came out (Dunsing et al., 2018) explaining how the assumption that all fluorophores emit signal is invalid and that because of this, oligomeric state changes calculated from brightness must be adjusted by a correction factor specific to the fluorophore. Unfortunately, this study did not characterize mClover, so we do not know its correction factor. I suspect that applying this correction would bring our figure of 1.6 a lot closer to 2. We also tried the naive choice of $\tau = 10$, which gave a decrease in brightness, showing that arbitrary parameter choices lead to unpredictable and unreliable results and should hence be avoided.

4.2 *In vitro* number and brightness

In our research group, we believe that the most practical quantitative method for measuring homo-dimerization *in vivo* and *in vitro* is N&B (Digman et al., 2008) because it is calibration-free and does not require specialised instrumentation. There are many examples of the application of N&B *in vivo* (the original N&B paper has over 250 citations, most of which are *in vivo* applications) but none *in vitro*. Hence, we published a protocol (Nolan et al., 2018) detailing how N&B can

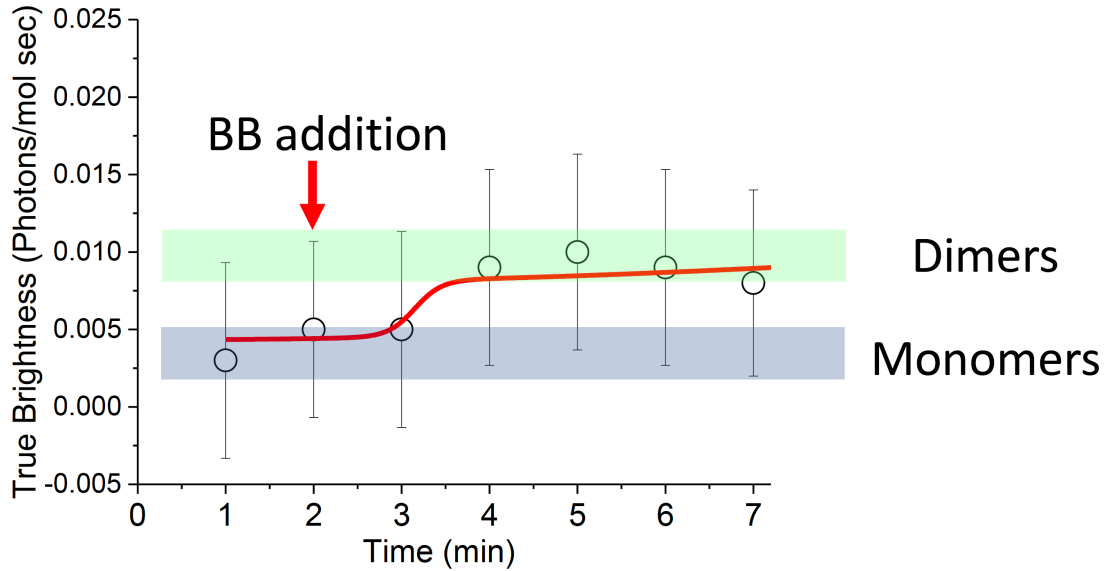


Figure 4.1: Dimerization of FKBP12F36V upon BB addition is seen by a brightness doubling from $\epsilon = 0.005$ to $\epsilon = 0.010$ over a period of minutes.

be applied *in vitro*. This time, we used FKBP12F36V which is an FKBP mutant with a new dimerizing drug AP20187 (known colloquially as the *BB dimerizer*); this pair is designed to have better specificity than the original (Clackson et al., 1998).

In this experiment, the FKBP12F36V was labelled with mVenus. We found that the brightness doubled from $\epsilon = 0.005$ to $\epsilon = 0.010$ upon addition of the drug. See figure 4.1. This analysis was done with exponential smoothing detrending with automatically chosen parameter τ . Without detrending, the pre-BB brightness was calculated as $\epsilon = 0.026$, showing that detrending is absolutely necessary and that neglecting this step can lead to nonsensical results.

Remark. This paper also included an important correction to the equation for brightness ϵ when analog equipment is used. The correct equation is

$$\epsilon = \frac{\sigma^2 - \sigma_0^2}{S(\langle I \rangle - \text{offset})} \quad (4.1)$$

The S in the denominator was omitted in the original paper ([Dalal et al., 2008](#)) and this error was reproduced in our N&B review ([Nolan et al., 2017b](#)).

4.3 HIV-1 receptor stoichiometry

4.3.1 Introduction

Our main motivation for studying N&B in the first place was that we thought it was a valuable method to study the process of HIV-1 fusion in live cells. N&B was first used in our research group to study the oligomeric state of dynamin at the HIV-1 fusion pore in TZM-bl cells ([Jones et al., 2017](#)). This study concluded that dynamin-2 stabilizes the HIV-1 fusion pore with a low oligomeric state.

Following on from this, we wanted to study the stoichiometry of the interaction of HIV-1 with its receptor (CD4) and co-receptor (CCR5 or CXCR4) upon the

engagement of the virus with the cell and to follow this interaction stoichiometry up to the point of fusion. See figure 4.2. See [Jakobsdottir et al. \(2017\)](#) for a review of HIV-1 fusion.

Entry of HIV-1 into a host cell requires an initial interaction between the viral-envelope glycoprotein spike complex—Env—with cell surface displayed CD4 and co receptors ([Jakobsdottir et al., 2017](#)). Although structural studies have revealed the intra-molecular basis for CD4 receptor and CXCR4/CCR5 co-receptor-induced conformational changes to the HIV-1 Env during host cell entry ([Ozorowski et al., 2017](#)), little is known about how the inter-molecular dynamics and stoichiometry of this process culminates in fusion with the host cell membrane in live cells ([Brandenberg et al., 2015a](#)). This is due to the difficulty of working with live cells and the lack of temporal resolution of the techniques commonly employed (i.e. crystallography and cryo-EM).

— [Iliopoulou et al. \(2018\)](#)

We saw N&B as the ideal technique to probe this stoichiometry temporally. With our microscope, we could acquire 100 frames per 1.7 minutes, therefore, using each consecutive sequence of 100 frames to create a brightness image, we could

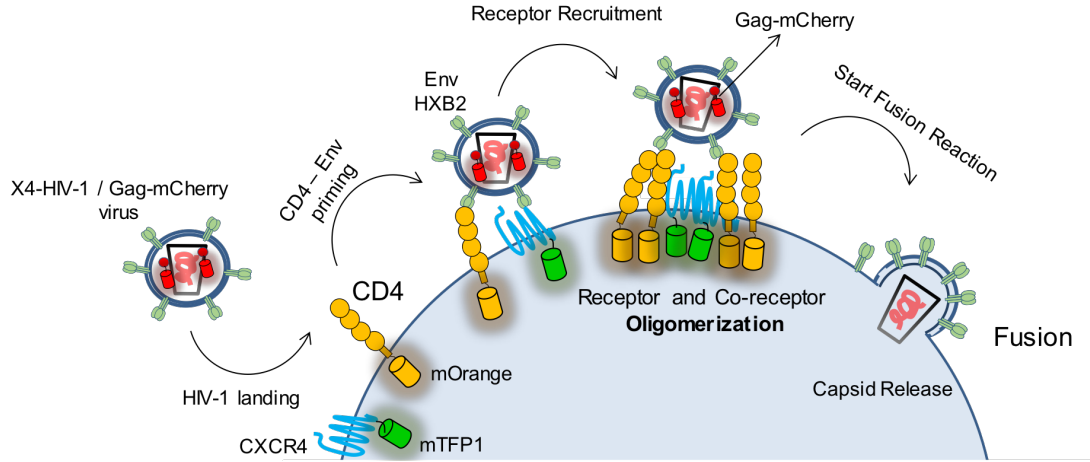


Figure 4.2: The HIV-1 envelope glycoprotein Env must bind the receptor (CD4) and form a complex with the co-receptor (CXCR4 or CCR5, this figure shows an X4-tropic virus and co-receptor) to initiate the fusion process. Labelling the viral Gag protein with mCherry, the receptor with mTFP1 and the co-receptor with mOrange, it is possible to follow these three players in the fusion reaction and to quantify their interaction.

obtain 1 brightness image every 1.7 minutes and use this to calculate this temporal stoichiometry.

4.3.2 Experimental setup

Receptor (mOrange) and co-receptor (mTFP1) were labelled in Cos7 cells. Virus was added at time $t = 0$ and imaging proceeded for a number of minutes at 100 frames per 1.7 minutes. Alternating laser excitation (ALEX, [Kapanidis et al. \(2005\)](#)) was used to eliminate the possibility of channel bleed-through. See figure [4.3](#).

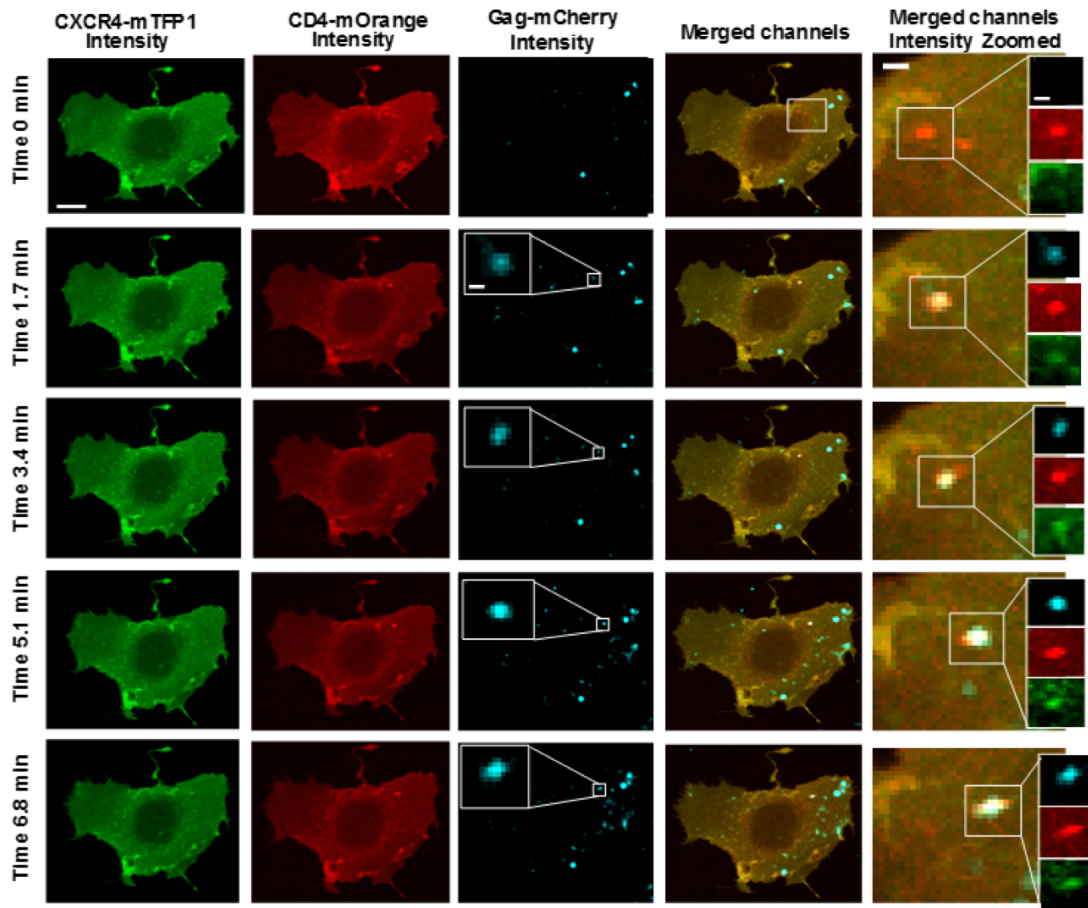


Figure 4.3: Intensity images from the virus, receptor and co-receptor. Every 100th frame is shown. A virus which lands at $t \approx 1.7$ minutes is highlighted.

4.3.3 Analysis

The virus channel was used to locate the virus at a given point in time. The receptor and co-receptor were used to calculate brightness and cross-correlated brightness every 100 frames (every 1.7 min). The brightness was used to determine the number of receptor and co-receptor units involved in a complex. The cross-correlated brightness was used to delimit whether or not the receptor and co-receptor units were together in the *same* complex. See figure 4.4, this is the corresponding brightness and cross-correlated brightness image of figure 4.3. Notice that once the virus lands, the oligomeric state of the receptor and co-receptor increases. We also see significant positive cross-correlated brightness in this area, indicating that the virus has triggered a complex of receptor and co-receptor.

4.3.4 Results

Figure 4.5 shows the results of the analysis detailed in figures 4.3 and 4.4 for $n = 10$ cases where virus triggered receptor and co-receptor complexes in the X4-tropic setting and $n = 12$ in the R5-tropic setting. A three-step pre-fusion process is hypothesized for each.

Our studies support a dynamic three step model for both HIV_{HXB2} and

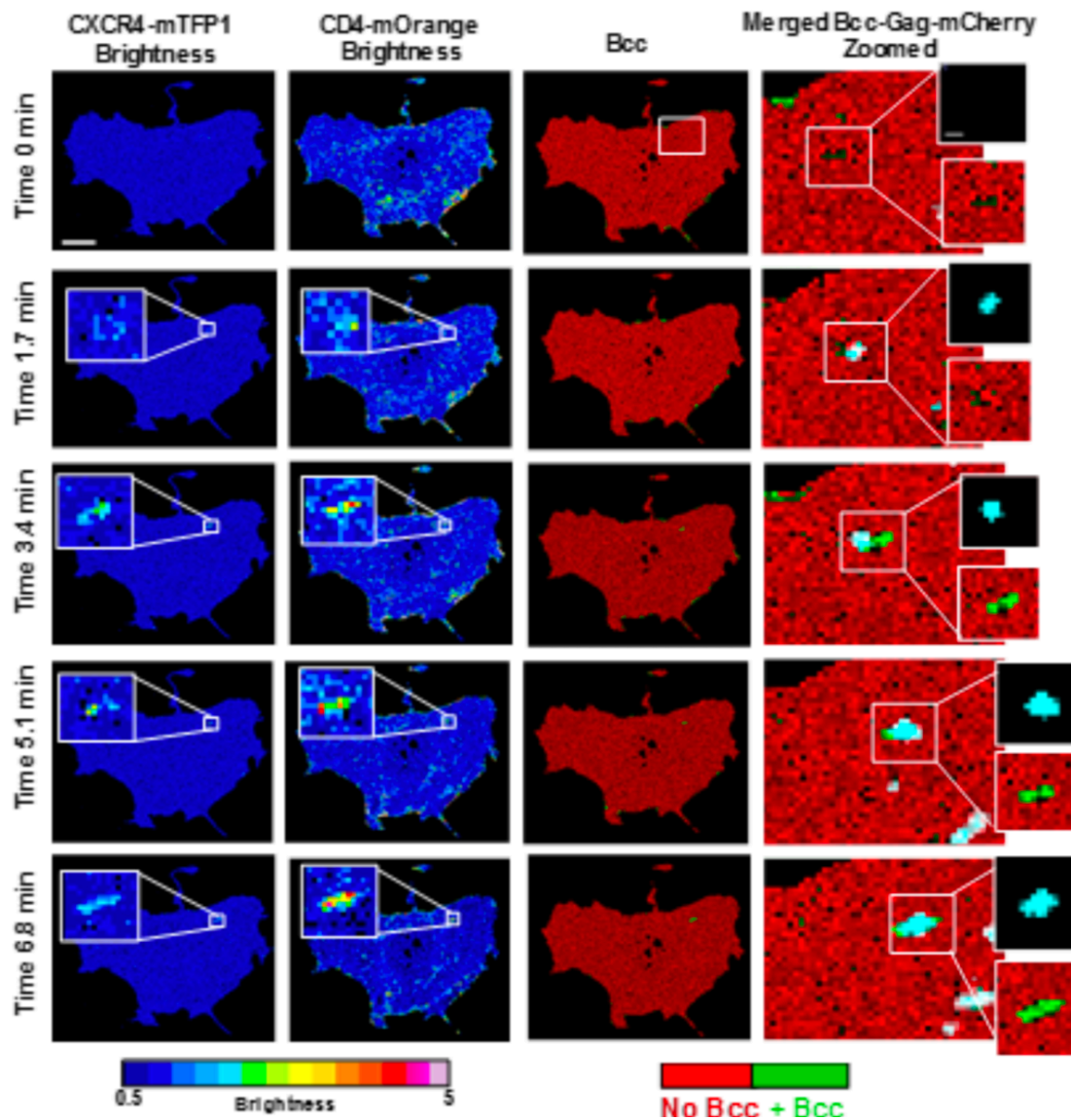


Figure 4.4: Brightness images of receptor and co-receptor and cross-correlated brightness image of the interaction between the two.

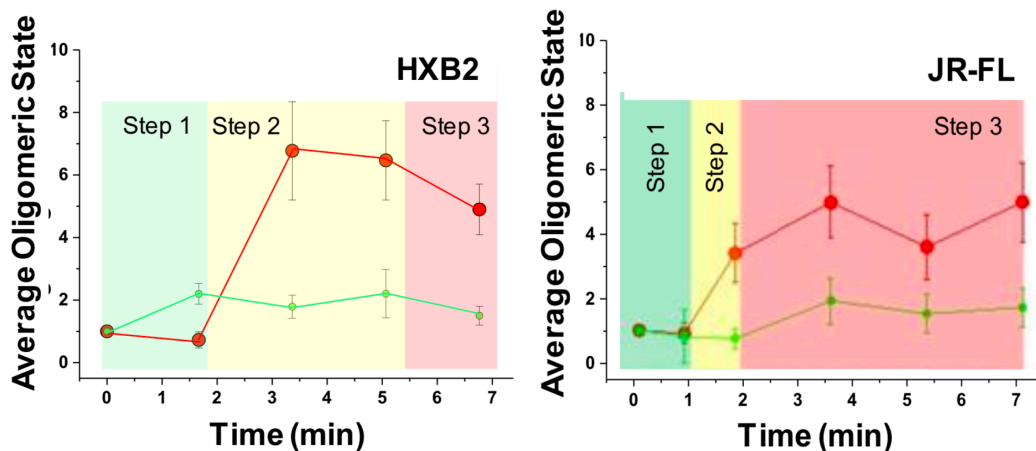


Figure 4.5: Number of receptor and co-receptor units involved in complexes with virus over time, obtained by brightness analysis.

HIV_{JR-FL} (figure 4.5). For X4 tropic virions, Env – CD4 interactions induce CXCR4 dimerization, CD4 then engages with two Env, - as shown by 3 color TIRF-dSTORM microscopy - to generate a hexamer that might serve as a scaffold to stabilise a final 4CD4 – 1/2CXCR4 conformation, with a single Env. We speculate that for HIVHXB2, step 2 is crucial to culminate the fusion reaction and there could be an anchoring domain and a fusion domain that undergoes gp120 disassembly leading to 6 helix bundle formation. For R5 tropic virions, Env – CD4 interactions form a the previously described asymmetric pre-hairpin intermediate; following binding and oligomerisation of 2 additional CD4 molecules with concomitant CCR5 dimerization. After this, the secondary intermediate leads to the final fusion competent complex with a total of 4 ± 0.3 CD4, 2 ± 0.3 CCR5 and 1 JR-FL Env.

— [Iliopoulou et al. \(2018\)](#)

For the X4-tropic case, some structural modelling of the hypothesis has been done. See figure [4.6](#).

4.3.5 Conclusion

Time-resolved N&B enabled us to answer questions about the interaction between the HIV-1 virus and its receptor and co-receptor in live cells which up to now could not be answered. Being able to correctly correct for bleaching is crucial for reliable N&B analysis, so the development of these new algorithms was crucial to this study.

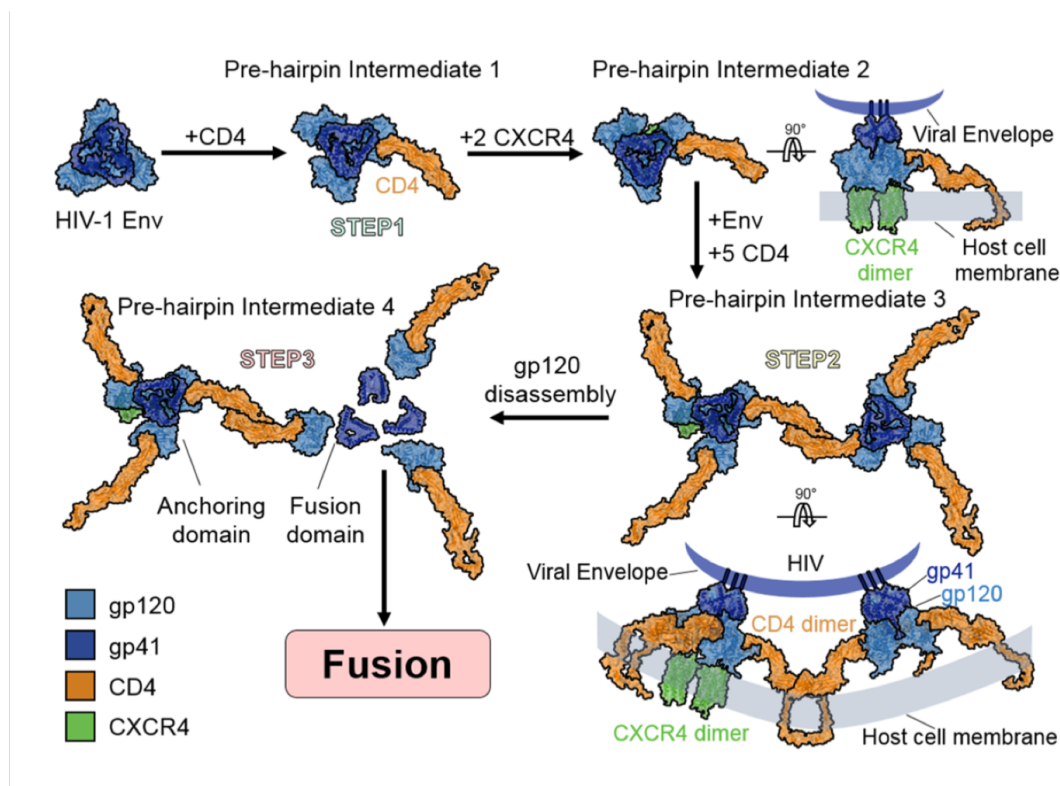


Figure 4.6: The HIV-1 JR-FL Env glycoprotein (Blue) (PDB ID: 4ZMJ) is present as a trimer on the mature virion. STEP 1: The initial attachment of the cognate host receptor, CD4 (Orange) to HIV-1 Env (PDB: 5VN3) forms an asymmetric pre-hairpin intermediate. STEP 2: Subsequent binding and oligomerisation of additional CD4 molecules on the trimer forms the secondary intermediate. STEP 3: The formation of this penultimate intermediate allows binding and dimerization of CCR5 co-receptors (green) results in the final fusion competent complex, with a total of 4 CD4 and 2 CCR5 molecules bound to a single HIV-1 Env trimer. b, Time-resolved stoichiometry pre-fusion reaction for CXCR4-CD4 (red dots) induced by HIVHXB2–Gag-iCherry virions. The HIV-1 Env glycoprotein is present as a trimer on the mature virion. Cognate host receptors, CD4 and CXCR4, exist as monomers and dimers on the cell surface prior to receptor recognition. The gp120 and gp41 subunits of the Env spike are colored light and dark blue, respectively. CD4 is colored orange. Attachment of the HIV-1 Env spike induces the formation of CXCR4 dimers (STEP 1) also seen in the literature (Kwon et al., 2015). CD4 clustering is likely facilitated by homodimerisation in the cytoplasmic domain, which enables cross-linking of at least two Env trimers (STEP 2) also seen in the literature (Brandenberg et al., 2015b). Normal brightness analysis reveals the limited degradation of these higher order receptor clusters over time (2 – 5 mins). We hypothesize that 1 Env plays the role of the anchoring and signaling domain whilst the other constitutes the fusion domain. Disassembly of gp120 in the fusion domain initiates the six helix bundle formation (STEP 3) and leads to the final 4CD4 – 2CXCR4 conformation in the anchoring domain. (gp120: light blue, gp41: dark blue, CD4: orange, CXCR4: green, gp120: light blue, gp41: dark blue, CD4: orange, CXCR4: green, gp120: light blue, gp41: dark blue, CD4: orange, CXCR4: green)

Chapter 5

Future plans

5.1 Robin Hood publication

The *Robin Hood* algorithm is already incorporated in the R package `detrendr` but it hasn't been published or peer-reviewed. Getting this done is my top priority.

5.2 Translate software to *ImageJ*

Whilst I really like R, the fluorescence community does not use it, which has been a major barrier to the use of my algorithms. I will code my detrending algorithms

as *ImageJ* plugins (and perhaps also as python modules) so that more of the community have easy access to them.

Bibliography

- Amara, J. F., Clackson, T., Rivera, V. M., Guo, T., Keenan, T., Natesan, S., Pollock, R., Yang, W., Courage, N. L., Holt, D. A., and Gilman, M. (1997). A versatile synthetic dimerizer for the regulation of protein-protein interactions. *Proceedings of the National Academy of Sciences*, 94(20):10618–10623.
- Brandenberg, O. F., Magnus, C., Regoes, R. R., and Trkola, A. (2015a). The HIV-1 entry process: A stoichiometric view. *Trends in Microbiology*, 23(12):763–774.
- Brandenberg, O. F., Magnus, C., Rusert, P., Regoes, R. R., and Trkola, A. (2015b). Different infectivity of HIV-1 strains is linked to number of envelope trimers required for entry. *PLoS Pathogens*, 11(1):e1004595.
- Chan, K. H., Hayya, J. C., and Ord, J. K. (1977). A note on trend removal methods: The case of polynomial regression versus variate differencing. *Econometrica*, 45(3):737.

- Chen, Y., Muller, J. D., Berland, K. M., and Gratton, E. (1999). Fluorescence fluctuation spectroscopy. *Methods*, 19(2):234–252.
- Clackson, T., Yang, W., Rozamus, L. W., Hatada, M., Amara, J. F., Rollins, C. T., Stevenson, L. F., Magari, S. R., Wood, S. A., Courage, N. L., Lu, X., Cerasoli, F., Gilman, M., and Holt, D. A. (1998). Redesigning an FKBP-ligand interface to generate chemical dimerizers with novel specificity. *Proc. Natl. Acad. Sci. U.S.A.*, 95(18):10437–10442.
- Dalal, R. B., Digman, M. A., Horwitz, A. F., Vetri, V., and Gratton, E. (2008). Determination of particle number and brightness using a laser scanning confocal microscope operating in the analog mode. *Microsc. Res. Tech.*, 71(1):69–81.
- Digman, M. A., Dalal, R., Horwitz, A. F., and Gratton, E. (2008). Mapping the number of molecules and brightness in the laser scanning microscope. *Biophys. J.*, 94(6):2320–2332.
- Digman, M. A. and Gratton, E. (2009). Imaging barriers to diffusion by pair correlation functions. *Biophys. J.*, 97(2):665–673.
- Digman, M. A., Wiseman, P. W., Choi, C., Horwitz, A. R., and Gratton, E. (2009). Stoichiometry of molecular complexes at adhesions in living cells. *Proceedings of the National Academy of Sciences*, 106(7):2170–2175.
- Dunsing, V., Luckner, M., Zühlke, B., Petazzi, R., Herrmann, A., and Chiantia,

- S. (2018). Quantifying protein oligomerization in living cells: A systematic comparison of fluorescent proteins.
- Hebert, B., Costantino, S., and Wiseman, P. W. (2005). Spatiotemporal image correlation spectroscopy (STICS) theory, verification, and application to protein velocity mapping in living CHO cells. *Biophys. J.*, 88(5):3601–3614.
- Iliopoulou, M., Nolan, R., Alvarez, L., Watanabe, Y., Coomer¹, C. A., Jakobsdottir, G. M., Bowden, T. A., and Padilla-Parra, S. (2018). A dynamic three step mechanism drives the HIV-1 prefusion reaction. *Nat. Struct. Mol. Biol.*, 0(0).
- Jakobsdottir, G. M., Iliopoulou, M., Nolan, R., Alvarez, L., Compton, A. A., and Padilla-Parra, S. (2017). On the whereabouts of HIV-1 cellular entry and its fusion ports. *Trends in Molecular Medicine*, 23(10):932–944.
- Jameson, D. M., Ross, J. A., and Albanesi, J. P. (2009). Fluorescence fluctuation spectroscopy: ushering in a new age of enlightenment for cellular dynamics. *Biophys Rev*, 1(3):105–118.
- Jones, D. M., Alvarez, L. A., Nolan, R., Ferriz, M., Urruela, R. S., Massana-Muñoz, X., Novak-Kotzer, H., Dustin, M. L., and Padilla-Parra, S. (2017). Dynamin-2 stabilizes the hiv-1 fusion pore with a low oligomeric state. *Cell reports*, 18(2):443–453.
- Kapanidis, A. N., Laurence, T. A., Lee, N. K., Margeat, E., Kong, X., and Weiss,

- S. (2005). Alternating-laser excitation of single molecules. *Accounts of Chemical Research*, 38(7):523–533.
- Kwon, Y. D., Pancera, M., Acharya, P., Georgiev, I. S., Crooks, E. T., Gorman, J., Joyce, M. G., Guttman, M., Ma, X., Narpala, S., Soto, C., Terry, D. S., Yang, Y., Zhou, T., Ahlsen, G., Bailer, R. T., Chambers, M., Chuang, G.-Y., Doria-Rose, N. A., Druz, A., Hallen, M. A., Harned, A., Kirys, T., Louder, M. K., O'Dell, S., Ofek, G., Osawa, K., Prabhakaran, M., Sastry, M., Stewart-Jones, G. B. E., Stuckey, J., Thomas, P. V., Tittley, T., Williams, C., Zhang, B., Zhao, H., Zhou, Z., Donald, B. R., Lee, L. K., Zolla-Pazner, S., Baxa, U., Schön, A., Freire, E., Shapiro, L., Lee, K. K., Arthos, J., Munro, J. B., Blanchard, S. C., Mothes, W., Binley, J. M., McDermott, A. B., Mascola, J. R., and Kwong, P. D. (2015). Crystal structure, conformational fixation and entry-related interactions of mature ligand-free HIV-1 env. *Nature Structural & Molecular Biology*, 22(7):522–531.
- Laboratory for Fluorescence Dynamics (2018). *Globals for Images: SimFCS 4*.
- Landini, G., Randell, D., Fouad, S., and Galton, A. (2016). Automatic thresholding from the gradients of region boundaries. *Journal of Microscopy*, 265(2):185–195.
- Nolan, R., Alvarez, L., Elegheert, J., Iliopoulou, M., Jakobsdottir, G. M., Rodriguez-Muñoz, M., Aricescu, A. R., and Padilla Parra, S. (2017a). nandb—

- number and brightness in r with a novel automatic detrending algorithm. *Bioinformatics*, 33(21):3508–3510.
- Nolan, R., Alvarez, L., Griffiths, S. C., Elegheert, J., Siebold, C., and Padilla-Parra, S. (2018). Calibration-free in-vitro quantification of protein homooligomerization using commercial instrumentation and free, open source brightness analysis software. *Journal of Visualized Experiments*, 0(0).
- Nolan, R., Iliopoulou, M., Alvarez, L., and Padilla-Parra, S. (2017b). Detecting protein aggregation and interaction in live cells: A guide to number and brightness. *Methods*.
- Nolan, R. and Padilla-Parra, S. (2017a). examplestr - an easy start to unit testing r packages. *Wellcome Open Research*, 2:31.
- Nolan, R. and Padilla-Parra, S. (2017b). filesstrings: An r package for file and string manipulation. *The Journal of Open Source Software*, 2(14):260.
- Nolan, R. and Padilla-Parra, S. (2018). ijtiff: An r package providing TIFF i/o for ImageJ users. *Journal of Open Source Software*, 3(23):633.
- Ozorowski, G., Pallesen, J., de Val, N., Lyumkis, D., Cottrell, C. A., Torres, J. L., Copps, J., Stanfield, R. L., Cupo, A., Pugach, P., Moore, J. P., Wilson, I. A., and Ward, A. B. (2017). Open and closed structures reveal allostery and pliability in the HIV-1 envelope spike. *Nature*, 547(7663):360–363.

- Qian, H. and Elson, E. L. (1990). Distribution of molecular aggregation by analysis of fluctuation moments. *Proc. Natl. Acad. Sci. U.S.A.*, 87(14):5479–5483.
- R Core Team (2016). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria.
- Rueden, C. T., Schindelin, J., Hiner, M. C., DeZonia, B. E., Walter, A. E., Arena, E. T., and Eliceiri, K. W. (2017). ImageJ2: ImageJ for the next generation of scientific image data. *BMC Bioinformatics*, 18(1):529.
- Schindelin, J., Arganda-Carreras, I., Frise, E., Kaynig, V., Longair, M., Pietzsch, T., Preibisch, S., Rueden, C., Saalfeld, S., Schmid, B., Tinevez, J. Y., White, D. J., Hartenstein, V., Eliceiri, K., Tomancak, P., and Cardona, A. (2012). Fiji: an open-source platform for biological-image analysis. *Nat. Methods*, 9(7):676–682.
- Stroustrup, B. (2013). *The C++ Programming Language*. Addison-Wesley Professional, 4th edition.