Epidemiological Motivation of the Transmission Risk Level

CWA Team

2020-06-15

Contents

1 Abstract

This document contains an epidemiological description of the *transmission risk level* used in the German Corona-Warn-App (CWA). As its name suggests, the transmission risk is an essential part when estimating the overall risk of a person to get infected in an exposure incident. Usage of the transmission risk level is specified in the ExposureNotification API and in the CWA Architecture. In particular we use epidemiological information about COVID-19 from the literature to motivate the choice of levels for this parameter. To enhance transparency and reproducibility of the computations, we provide the mathematical derivations and the computations in one Rmarkdown document. The methods sketched below are likely to be subject to change, once additional information about the characteristics of COVID-19 is obtained or as feedback from the use of the app arrives.

2 Introduction

We are interested in the situation where a person **A** (potential infector) at time t_0 uploads information about being a laboratory confirmed SARS-CoV-2 case. The upload happens in terms of A's *diagnosis keys* (see Apple and Google 2020b). Each *diagnosis key* is associated with a particular day in A's history (past 14 days) and also has an optional *transmission risk level* from I–VIII (see Apple and Google 2020a).

Users can periodically download *diagnosis keys* from the *diagnosis server*. For each app user **B** (potential infectee), who downloads the list of valid *diagnosis keys* and discovers he or she has been in contact with *A*, a risk assessment will be made.[1] This risk assessment is operationalized by the *total risk score*, of which one component is the *transmission risk level* λ_A computed by *A*. The *transmission risk level* (provided by *A*) and its associated *transmission risk value* (set by *B*) are app-defined and should be based on the probability of transmission between the two persons being in close contact. In the present document we interpret this probability as a function of epidemiological information about *A* and the time of contact. Information about the closeness and the duration of the contact are not considered part of the transmission risk component, because they are handled separately in the computation of the *total risk score*. For more information on how to calculate the *total risk score* see the Exposure Notification API.

As the *transmission risk level* is computed on *A*'s device, additional information about *A* such as *being symptomatic or not*, *date of onset of symptoms*, *date of sampling* or *date of test result* could be used to estimate the **infectiousness of A** more precisely. We will use the currently known characteristics of COVID-19, especially its infectiousness profile due to viral shedding and the operational delays of its handling to estimate the infectiousness at certain times. We can obtain the information on how many days ago from **now** (i.e. t_0) the contact between *A* and *B* was: Let t_C be the time of contact between *A* and *B*, then the contact was $d = t_0 - t_C$ days ago. The aim of this document is thus to parametrize the time-dependent infectiousness of *A* as a function of d.

The better we can assess the probability of a transmission from *A* to *B*, the more accurate is the *combined risk score* that is used to warn the user to take further action, e.g., to contact a local health authority. Having digital support for this type of contact tracing appears helpful in order to obtain a more complete coverage of contact tracing and to do this much quicker.

The present document is structured as follows. In section Scenarios we distinguish between four possible information states about *A* at the time of upload[2] depending on whether an onset of COVID-19 symptoms has occurred or not and whether this information can be used or not. We calculate a transmission risk for each of the four cases. However, since the initial version of the app will not allow a distinction between cases 1–4, and since the *transmission risk level* is only one of 4 components for the *total risk score*, we thus normalize the *transmission risk level* in a so called base case, which will be used by the initial version of the app. A section Discussion summarizes the results and points out important limitations.

3 Scenarios and Events

Since the API allows for a customization of the transmission component λ_A in the above, we shall study it in more detail here. Particular interest will be in four information scenarios about *A*, the potential infector, at the time of the upload.

For an infected person the following sequence of event times occurs (but not necessarily in the given order):

- $T_E = T_{\text{infection}}$: transmission of SARS-CoV-2 to an **e**xposed person *A* from some unknown source
- $T_I = T_{\text{infectious}}$: start of the infectious period in person *A*, i.e. *A* is able to infect others
- $T_S = T_{\text{symptoms}}$: onset of symptoms in person *A* (also referred to as **DSO**, day of symptom onset)
- $T_P = T_{\text{sampling}}$: time of sampling of person *A* [dt. **P**robenentnahme]
- $T_R = T_{\text{result}}$: time of *A* obtaining the positive test result
- $T_U = T_{\text{upload}}$: time where person *A* uploads the positive test result to the system

Note that before observation these times are to be considered as random quantities and, hence, are denoted by an uppercase letter. Once observed, a lower case letter shall be used.

---

[1] Throughout we will assume that the contact between *A* and *B* was such that *A* infected *B*. At this point we explicitly ignore the possibility that *B* actually infected *A*.

[2] Due to technical restrictions, the upload of the diagnosis keys may not be a single-point-in-time event, but may happen over two consecutive days, without further interaction with the user after his or her initial consent. Given that the time point of the user's consent for upload may also be the last opportunity for providing additional information such as date of onset of symptoms, we use "consent for upload" when calculating delays and refer to it as simply "upload" throughout this document.

Furthermore, note that for SARS-CoV-2 it is likely that T_I occurs before T_S, but it cannot be ruled out that the order is reversed. For asymptomatic cases T_S will never occur, but T_I might already have occurred or will occur in the future. For pre-symptomatic cases T_S lies in the future, i.e. $T_S > t_0$, but T_I might already have occurred or lie in the future. From our above description we would usually have $T_{\text{upload}} = t_0$. In a later section we will study the delays between the different event times. For this we will have to use the convolution of random variables to get the distribution of their sum.

To derive the transmission component λ_A we will distinguish persons who will eventually develop symptoms and those which are completely asymptomatic. The former set will be denoted $\mathcal{S}ymp$ and the later $\mathcal{A}symp$. Throughout the text we will use the shorthand notation $\mathcal{A}symp_A$ to denote the event that person A belongs to the set of completely asymptomatic. Likewise for $\mathcal{S}ymp_A$. We reserve the notion of *asymptomatic* for those persons who never develop symptoms, whereas *pre-symptomatic* at a particular time t are those who will eventually develop symptoms, but at a later point in time than t. If T_S^A denotes the time of symptom onset in A, we will use the shorthand notation $\mathcal{S}ymp_A(T_S^A > t)$ to denote the event that A belongs to the set of individuals who will eventually develop symptoms, but the onset of these symptoms has not yet occurred by time t, i.e.

$$\mathcal{S}ymp_A(T_S^A > t) = \{\omega \in \mathcal{S}ymp_A \mid \omega : T_S^A > t\}.$$

Likewise we define the event $\mathcal{S}ymp_A(T_S^A \leq t)$. Both asymptomatic and pre-symptomatic are characterized as being *non-symptomatic* at a given time of reference t. We will use the following shorthand to denote this event:

$$\mathcal{N}symp_A(T_S^A > t) = \mathcal{A}symp_A \cup \mathcal{S}ymp_A(T_S^A > t),$$

which is just a formal way of saying that in order for A to be non-symptomatic at time t, A either belongs to the set of completely asymptomatic cases or A is pre-symptomatic at time t. Then we have (under the assumption that A is infected)

$$
\begin{aligned}
1 &= P(\mathcal{S}ymp_A) + P(\mathcal{A}symp_A) \\
&= P(\mathcal{S}ymp_A(T_S^A \leq t) + P(\mathcal{S}ymp_A(T_S^A > t)) + P(\mathcal{A}symp_A) \\
&= P(\mathcal{S}ymp_A(T_S^A \leq t)) + P(\mathcal{N}symp_A(T_S^A > t))
\end{aligned}
$$

The four information scenarios about A are then:

1. symptomatic and day of symptom onset $t_S^A \leq t_0$ known at time t_0, i.e. the event $\mathcal{S}ymp_A(T_S^A = t_S^A)$,
2. symptomatic but unknown day of symptom onset at time t_0, i.e. the event $\mathcal{S}ymp_A(T_S^A \leq t_0)$,
3. non-symptomatic at time t_0, i.e. the event $\mathcal{N}symp_A(T_S^A > t_0)$,
4. no knowledge of symptom status at time t_0 (the *base case*), i.e. the event $\Omega = \mathcal{S}ymp_A(T_S^A \leq t_0) \cup \mathcal{N}symp_A(T_S^A > t_0)$.

Differentiation of these scenarios requires person A to provide additional (optional) information on potential symptom onset and respective date. The 2nd scenario occurs, if A accepts to reveal that COVID-19 relevant symptoms have been observed by the time of upload, but A does not want to reveal (or does not know) the day of symptom onset. If A despite a positive test result either has not yet developed or never will develop any symptoms, then we would be in scenario 3. If A does not provide any additional information, this would lead to scenario 4.

3.1 Infectiousness Profile due to Viral Shedding

Infectiousness of COVID-19, i.e. how much infectious material is being shed, varies as a function of time since infection, the development of symptoms and (if available) the DSO, see for example He et al. (2020). Assuming the amount of virus shed by a symptomatic case A is described by the function $v_A(d)$, where d is the days since DSO in A. We expect the function to be positive even for negative d values, due to pre-symptomatic transmission. Note that the scale of $v_A(d)$ is in principle arbitrary for our purposes as we are interested in the amount of virus shedding compared to the potential maximum value which informs the

eventual classification into the risk levels I to VIII. Note also that symptomatic cases with unknown date of onset and completely asymptomatic cases are not handled by the function $v_A(d)$. These will be addressed later in this section.

The study by He et al. (2020) examines the temporal dynamics of viral shedding and infectiousness of symptomatic COVID-19 cases. They provide results informing on the transmission risk for contacts of symptomatic cases.

Based on 77 identified transmission pairs He et al. (2020) inferred the infector's infectiousness profile with respect to symptom onset by fitting a left-shifted gamma-distribution to the empirical frequency of observed transmissions occurring at d days before or after symptom onset of the infector. The left-shifting of the gamma-distribution allows for potential pre-symptomatic transmission. The resulting infectious profile is shown in Fig. 1c (middle) in He et al. (2020), suggesting that infectiousness starts at 3 days prior to DSO and ends 8 days after DSO, with peak infectiousness at one day before DSO.

Thus, for the profile function $v_A(d)$ we implement a discretized version of the infectiousness profile as inferred by He et al. (2020). The discretized profile for each day d computes the mean infectiousness within $[d, d+1)$, for instance the value $v_A(-2)$ refers to the mean infectiousness within the interval $[-2, -1)$ days, i.e. with respect to time of symptom onset. The profile $v_A(d)$ is normalized such that the maximum infectiousness by day equals one, which occurs at $d = -1$.

```
##    -3    -2    -1     0     1     2     3     4     5     6     7     8     9
## 0.015 0.185 0.237 0.197 0.139 0.091 0.057 0.034 0.020 0.011 0.006 0.004 0.002
##    10    11    12
## 0.001 0.001 0.000
```

```r
d_infprofile <- c(
  "-3" = 0.015, "-2" = 0.185, "-1" = 0.237, "0" = 0.197, "1" = 0.139,
  "2" = 0.091, "3" = 0.057, "4" = 0.034, "5" = 0.02, "6" = 0.011,
  "7" = 0.006, "8" = 0.004, "9" = 0.002, "10" = 0.001, "11" = 0.001,
  "12" = 0
)
d_infprofile <- d_infprofile / max(d_infprofile)

ggplot_pmf(d_infprofile) +
  xlab("Time relative to symptom onset (Days) of infector") +
  ylab("relative infectiousness") +
  coord_cartesian(ylim = c(0, 1))
```
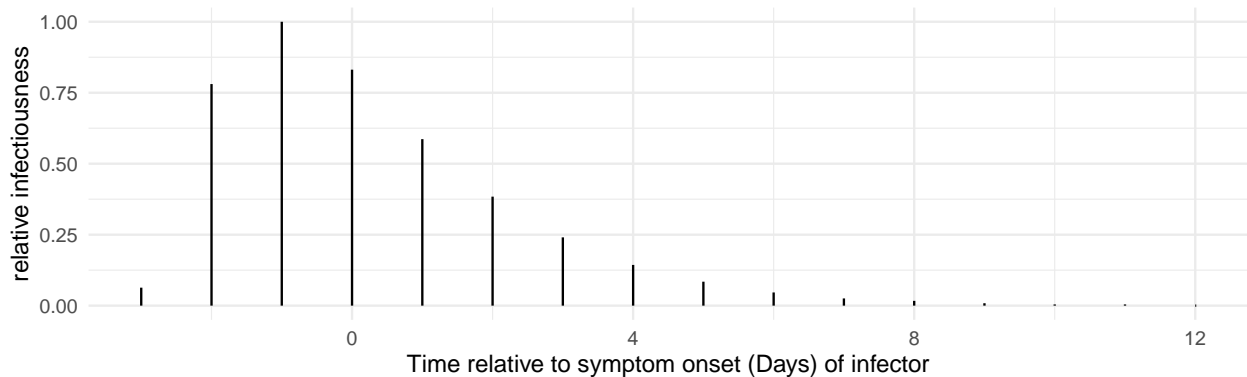

Figure 1: Assumed infectiousness profile.

It should be noted that the inferred infectious profile by He et al. (2020) has some bias, since the infectiousness profile represents the conditional probability density of the time point of transmission relative to the infector's

date of symptom onset, given that a transmission from A to B occurs. Thus, for transmission pairs with frequent contacts within a wide exposure window, this observed time of transmission merely represents the time of 'first successful transmission'. However, this eliminates the possibility for further transmissions (to be observed) between these persons, although the actual infectiousness of the infector might persist (and even further increase) after the observed transmission. The infectiousness was estimated to have completely vanished by 8 days after symptom onset, which might be just a consequence of this bias.

An indicator for this bias is the amount of virus shedding as a function of time since onset of symptoms, which is also provided within He et al. (2020). Virus shedding very likely corresponds well to the actual infectiousness of a symptomatic person and it was found that viral load only gradually decays until the end of the testing window (21 days after symptom onset), which suggests that infectiousness persists even beyond 8 days after symptom onset as inferred through the transmission pairs. However, a study by Wölfel et al. (2020) suggests, that persistent viral shedding beyond 8 days after symptom onset does not yield viable viruses that could be amplified in cell culture, and Arons et al. (2020) observed in a 'Skilled Nursing Facility' setting, that viable viruses were recovered more often in the days prior to symptom onset than in the subsequent days. Thus, the exact connection between the amount of shedding and the corresponding risk of transmitting the virus is still unclear.

Still, a more realistic infectiousness profile could be informed by also accounting for the duration of virus shedding, resulting in a longer duration of persisting infectiousness. Combining both of these data sources from He et al. (2020) with available and future data on virus viability in cell culture within a holistic statistical framework might yield such an adjusted infectiousness profile curve, which should be utilized in a future adoption of the transmission risk calculation.

3.2 Operational Delays

The time period between sampling until test result, i.e. $T_{\text{result}} - T_{\text{sampling}}$, is given by

```
d_samp2res <- c("0" = 0.1, "1" = 0.7, "2" = 0.2)
```

The time period between the test result and the upload, i.e. $T_{\text{upload}} - T_{\text{result}}$, is given by

```
d_res2upload <- c("0" = 0.7, "1" = 0.25, "2" = 0.05)
```

Note that the chosen probabilities in these two distributions are no more than educated expert guesses. They have to be adjusted based on real data once the system is running.

3.2.1 Sampling to Upload

The convolution of the time from (sampling to getting the result) and (getting the result to uploading) leads to the following distribution for the time between sampling and upload:

```
(d_samp2upload <- convolute(d_samp2res, d_res2upload))
```

```
##     0     1     2     3     4
## 0.070 0.515 0.320 0.085 0.010
```

```
ggplot_pmf(d_samp2upload) + xlab("Duration Sampling until Upload (Days)")
```
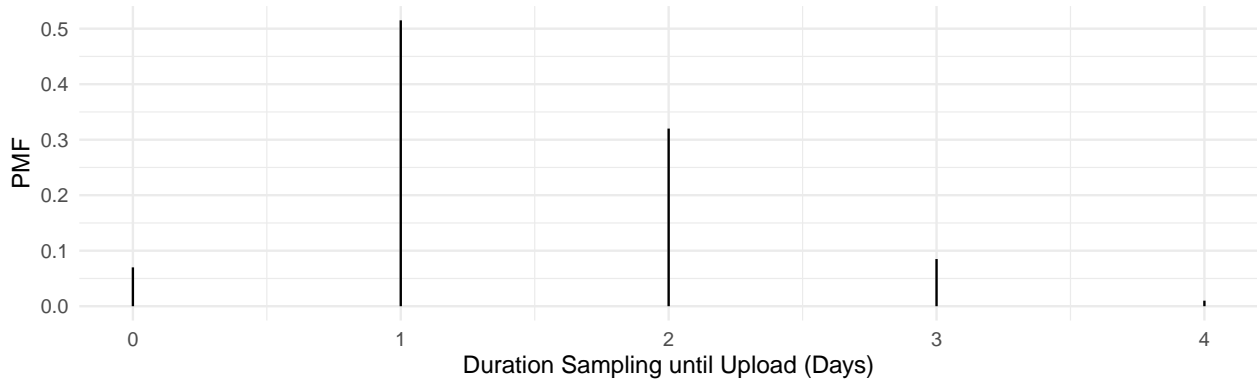
Figure 2: PMF of the duration between sampling and upload.

3.2.2 Symptoms to Upload for Symptomatic

To compute the delay between onset of symptoms and and upload we need a distribution for the time between symptom onset and time of sampling. Note that for this distribution we assume that the infected person gets tested *because* of the developed symptoms and thus this delay cannot be negative. For the alternative scenario, in which a person gets tested for other reasons, the onset of symptom might occur after taking a sample. This case is treated further below.

We assume $T_{\text{sampling}} - T_{\text{symptoms}}$ for symptomatic individuals is a discrete distribution with the following PMF:

```
d_symp2samp <- c("0" = 0.1, "1" = 0.6, "2" = 0.2, "3" = 0.1)
```

This leads to the convoluted time from DSO to upload displayed as follows:

```
(d_symp2upload <- convolute(d_symp2samp, d_samp2upload))
```

```
##      0      1      2      3      4      5      6      7
## 0.0070 0.0935 0.3550 0.3105 0.1675 0.0550 0.0105 0.0010
```

```
ggplot_pmf(d_symp2upload) + xlab("Duration Symptom Onset until Upload (Days)")
```
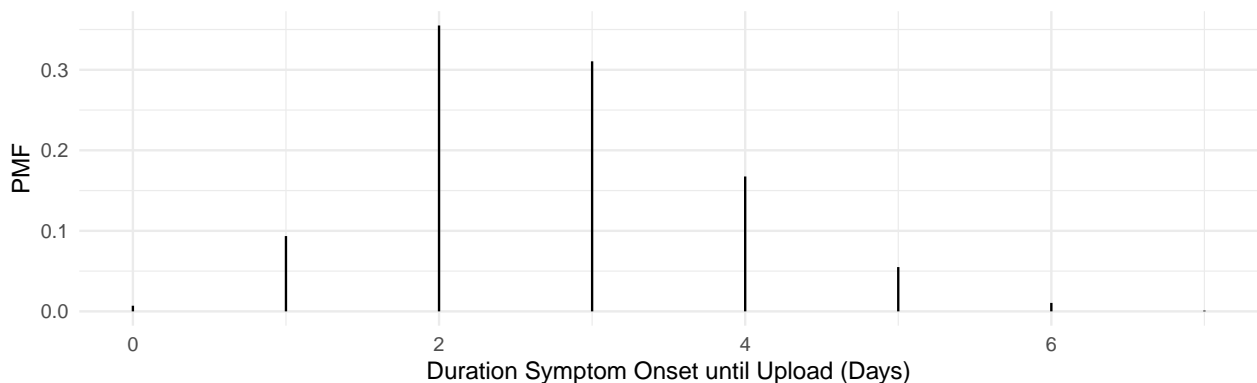

Figure 3: PMF of the duration between DSO and upload.

3.2.3 Upload to Symptoms for Pre-Syptomatic

Here, we cover the scenario in which people get tested independent of having developed any symptoms beforehand and in case of a positive test result, upload their diagnosis keys. This could, e.g., be the case

for individuals which are tested as part of contact tracing, because they were identified as a contact at risk after having had contact with an infected person. For these cases the time from symptom onset to upload may be in fact negative, since symptoms might begin after receiving the test result and upload. Note that we here condition on DSO actually occurring, which is not the case for truly asymptomatic cases.

To compute the delay distribution for this case we define the delay distribution from sampling to DSO for cases which were tested within their pre-symptomatic phase and will develop symptoms afterwards. Since a PCR-test result is most likely to be positive in a phase of considerable viral shedding, we assume that only tests within the pre-symptomatic phase will come out positive, which means that only tests from samples taken within the 3 days prior to DSO will be positive. Thus, as we have no further information when suspected cases get tested during their pre-symptomatic stage, we assume that it takes between 1 and 3 days from sampling to symptom onset, each with probability 1/3. In other words, the sampling occurs 1 to 3 days *prior* to DSO for pre-symptomatically tested. Thus the time from DSO to sampling is negative. We denote the distribution of this delay by `d_symp2samp_presymptomatic`.

```
(d_symp2samp_presymptomatic <- convolute(
  c("0" = 1 / 3, "1" = 1 / 3, "2" = 1 / 3), c("-3" = 1)))
```

```
##         -3         -2         -1
## 0.3333333 0.3333333 0.3333333
```

By convoluting the (negative) time delay from symptom onset to sampling `d_symp2samp_presymptomatic` with the delay from sampling to upload `d_samp2upload` we obtain the overall time delay from upload to symptom onset, denoted by `d_upload2symp`. Note that this delay may be negative or positive, since some cases, although tested in a pre-symptomatic stage, might have already developed symptoms at the time of upload.

```
d_upload2symp <- convolute(d_symp2samp_presymptomatic, d_samp2upload)
ggplot_pmf(d_upload2symp) + xlab("DSO relative to upload date (Days)")
```
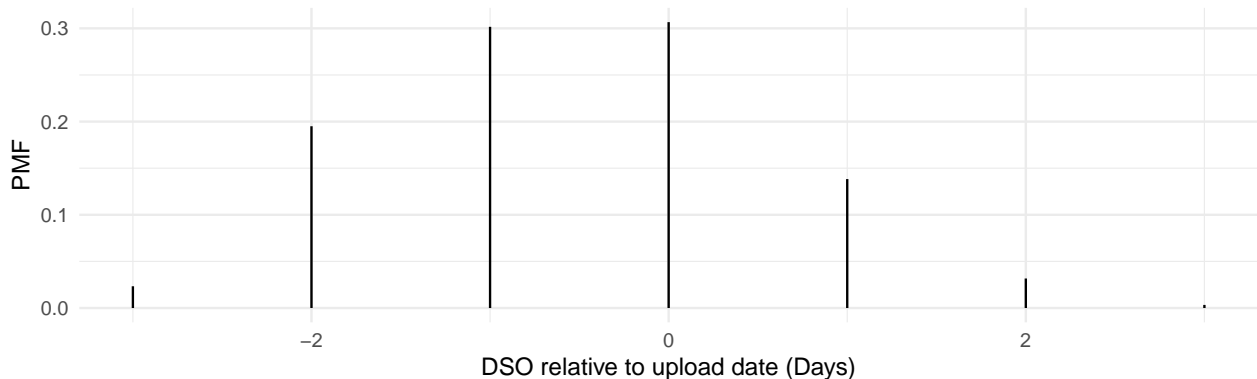

Figure 4: PMF of the duration between DSO and upload.

3.3 Raw Transmission Risk

Assume that we know that A got a positive test result, which is uploaded to the server on day t_0. All contacts that occurred $d = t_0 - t_C$ days ago will be assigned the same *transmission risk level* $\lambda_A(d)$, containing information on the infectiousness of a generic contact of A on that day. The infectiousness and therefore also the risk level $\lambda_A(d)$ might depend on further information provided by A, in particular whether A developed symptoms by the time of upload and in that case the day of symptom onset $T_S^A = t_S^A$, which would imply that $t_S^A \leq t_0$.

The *transmission risk level* $\lambda_A(d)$ is derived in two steps: 1. First, we compute the *raw relative infectiousness* (also referred to as *transmission risk*) $\lambda_A^{(raw)}(d)$ of A given the provided information, which serves as a

continuous infectiousness value on the scale $[0, 1]$. For this step we distinguish between the four possible scenarios of available information introduced in section Scenarios and Events. 2. In a second step, this raw value is translated into a *transmission risk level* $\lambda_A(d)$ which takes values from 1 to 8 and which will be further used within the *total risk score* calculation. This classification will be explained in section Transmission Risk Levels.

3.3.1   Case 1: Availability of t_S^A

If the time of symptom onset is available, i.e. in the event $\mathcal{S}ymp_A(T_S^A = t_S^A)$, then at time t_0, the onset of symptoms in the primary case A happened $t_0 - t_S^A \geq 0$ days ago. So the relative infectiousness of case A relative to t_0 is just $v_A(-d)$ shifted $t_0 - t_S^A$ days to the right, i.e $v_A(-d + (t_0 - t_S^A)) = v_A(t_C - t_S^A)$.

```r
# Infectious profile relative to t0
d_infprofile_t0 <- function(t0_minus_tS) {
  res <- d_infprofile
  names(res) <- as.numeric(names(res)) - t0_minus_tS
  res
}


plot_it <- function(t0_minus_tS, xlim) {
  d_infprofile_t0_diff1 <- d_infprofile_t0(t0_minus_tS = t0_minus_tS)
  ggplot_pmf(d_infprofile_t0_diff1) +
    xlab(expression("Time of exposure of secondary case relative to "
                    * t[0] * " and " * t[S]^A *
                      " in primary case known (Days)")) +
    ylab("relative infectiousness") +
    ggtitle(substitute(t[0] - t[S]^A == a, list(a = t0_minus_tS))) +
    xlim(xlim)
}
gridExtra::grid.arrange(plot_it(1, xlim = c(-5, 10)), plot_it(2, xlim = c(-5, 10)))
```
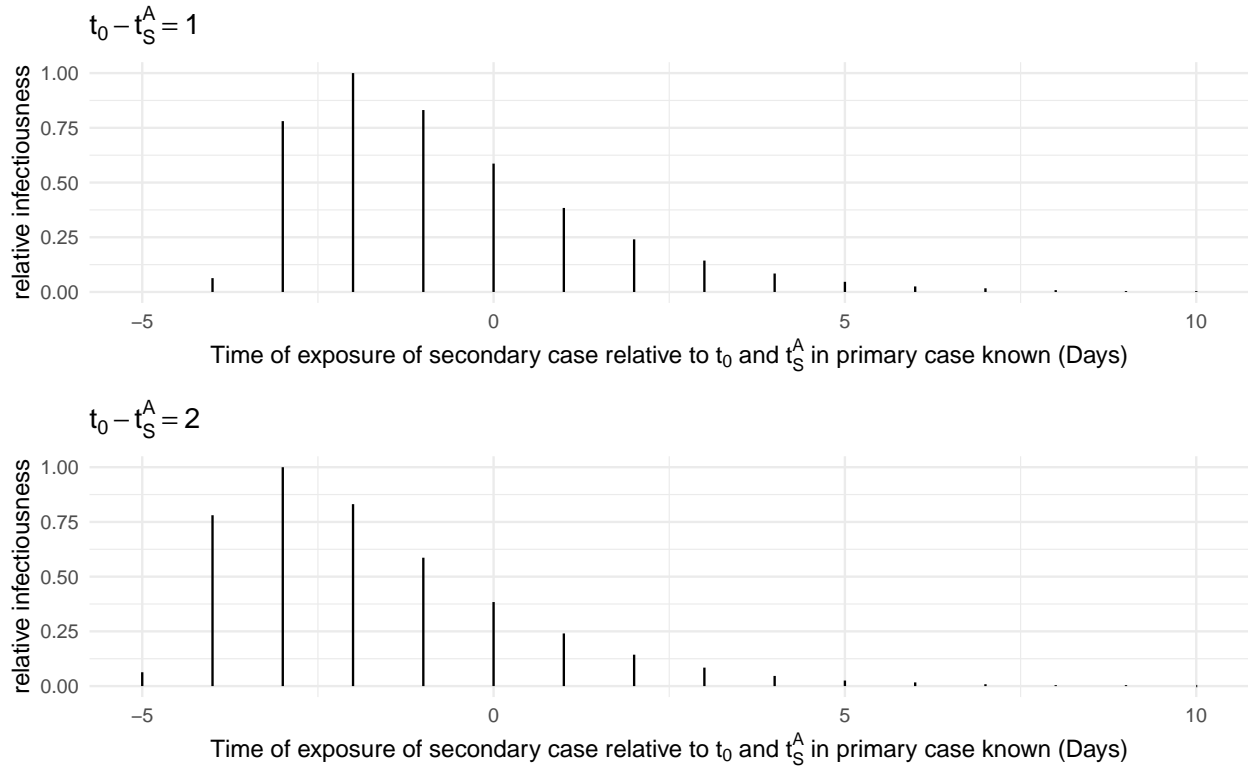
Figure 5: Plot of the infectiousness profile, if symptom onsets happened 1 (top) or 2 (bottom) days ago from upload.

Hence, for a given DSO t_S^A we define the raw transmission risk

$$\lambda_A^{(raw)}(d, \mathcal{S}ymp_A(T_S^A = t_S^A)) \ = \ v_a(-d + t_0 - t_S^A),$$

where $d \ = \ t_0 - t_C$ refers to the duration in days between the time of contact and the time of upload. Again, here we only provide the raw value $\lambda_A^{(raw)}$, i.e. the relative infectiousness on a $[0, 1]$-scale. These raw infectiousness values as a function of DSO and time since contact are shown below.

```
# Maximum number of days since exposure to display
max_dse <- 13

M_case1 <- matrix(
  0,
  nrow = 22,
  ncol = max_dse + 1,
  dimnames = list(days_since_symptoms = 0:21, days_since_contact = 0:max_dse)
)
for (i in 0:21) {
  inf_profile <- d_infprofile_t0(t0_minus_tS = i)
  days <- as.numeric(names(inf_profile))
  days_since_contact <- days * (-1)
  # Only pick events in the past since you condition on the two having met
  reasonable_days <- (days <= 0) & (days_since_contact <= max_dse)
  days_since_contact <- days_since_contact[reasonable_days]
  M_case1[i + 1, days_since_contact + 1] <- inf_profile[reasonable_days]
}
```
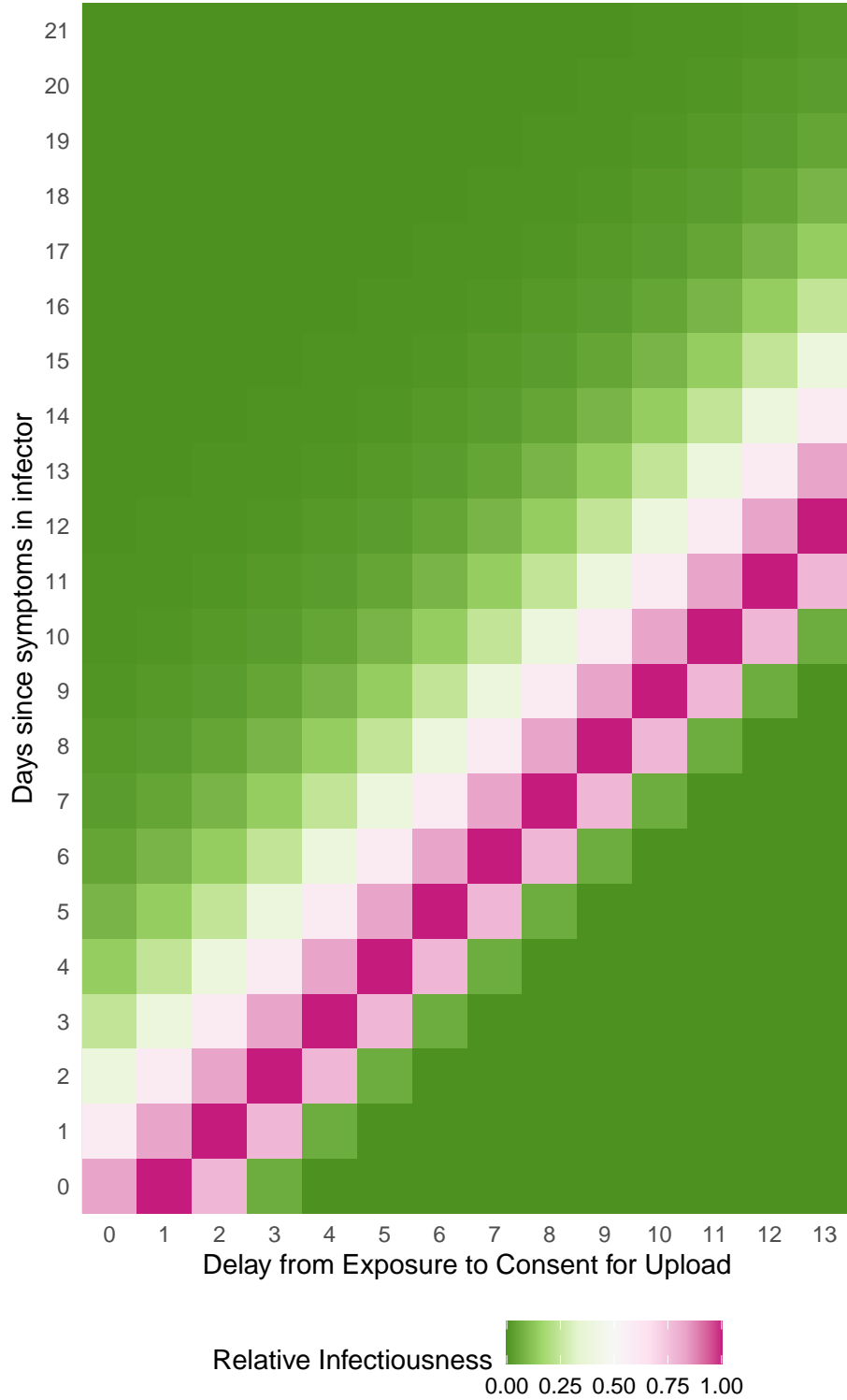
Figure 6: Raw transmission risk for Case 1: A is symptomatic with DSO provided.

3.3.2 Case 2: Symptomatic, but no Availability of t_S^A, only Day of Upload

Here we consider the case, in which we know that A is symptomatic at time of upload, but do not know the exact DSO, i.e. the event $\mathcal{S}ymp_A(T_S^A \leq t_0)$. In this case, we can infer a probability distribution for the DSO

T_S^A from our knowledge about the duration between DSO and upload given by `d_symp2upload` and knowing the date of upload t_0. We therefore define the raw transmission risk as the conditional expectation of $\lambda_A^{(raw)}$ given that A was symptomatic somewhere before t_0. We obtain this expectation by marginalizing over the possible DSO T_S^A, i.e.

$$\lambda_A^{(raw)}(d, \mathcal{S}ymp_A(T_S^A \leq t_0)) = \mathbb{E}_{T_S^A}\left[\lambda_A^{(raw)}(d, \mathcal{S}ymp_A(T_S^A = t_S^A)) \,\Big|\, \mathcal{S}ymp_A(T_S^A \leq t_0)\right]$$

$$= \sum_{t_S^A = t_0 - 13}^{t_0} \lambda_A^{(raw)}(d, \mathcal{S}ymp_A(T_S^A = t_S^A)) \cdot d_{\texttt{symp2upload}}(t_0 - t_S^A)$$

```
M_case2 <- numeric(max_dse + 1)
for (days_since_contact in 0:max_dse) {
  case1_val <- M_case1[as.numeric(names(d_symp2upload)) + 1, days_since_contact + 1]
  M_case2[days_since_contact + 1] <- sum(case1_val * d_symp2upload)
}

# Convert result to matrix for better comparison
M_case2 <- matrix(
  M_case2,
  ncol = max_dse + 1,
  nrow = 1,
  dimnames = list("no info about DSO at upload", days_since_contact = 0:max_dse)
)

# Convert to data.frame
df_M_case2 <- matrix_to_df(M_case2)

# Show the scale
plot_rel_infectiousness(df_M_case2)
```
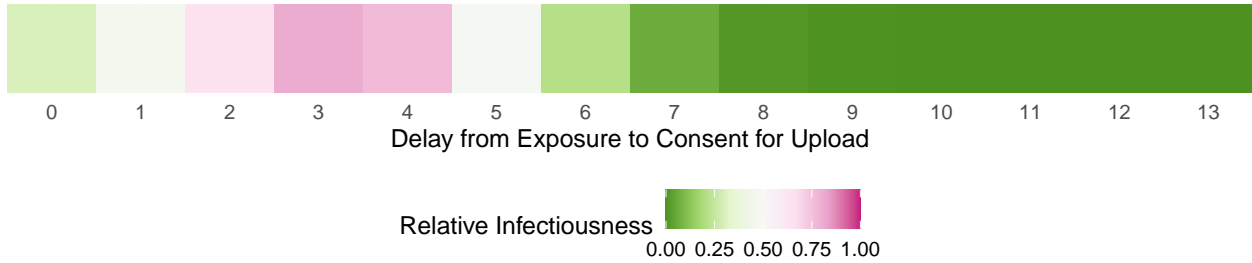

Figure 7: Raw transmission risk for Case 2: A is symptomatic with unknown DSO.

Due to marginalization, the resulting raw transmission value (as a function of $d = t_0 - t_C$) is flatter and wider compared to a raw transmission value curve for a known date of symptom onset t_S^A (see case 1). In particular, the maximum raw infectiousness value is reached at 0.81 compared to 1 in the case of a known DSO. This difference shows the additional gain in risk assessment within the case of knowing the DSO.

Note that for intermediate scenarios in which some additional information, like the date of sampling, is provided by A, one can compute the raw transmission value similarly by marginalizing out the date of symptom onset with respect to the corresponding delay distribution. This again could yield a more informative value compared to relying on the upload date alone.

3.3.3 Case 3: Completely Asymptomatic or Pre-Symptomatic

For this case we consider the scenario in which A at the time of upload provides the information that there was no onset of symptoms so far, i.e. the event $\mathcal{N}symp_A(T_S^A > t_0)$. This case implies the two mutually exclusive situations:

i. $\mathcal{S}ymp_A(T_S^A > t_0)$, i.e. A will develop symptoms at a later time point $T_S^A > t_0$ and is currently in a pre-symptomatic stage

ii. $\mathcal{A}symp_A$, i.e. A will not develop any symptoms at all

At the time of upload we will not know which of these two situations applies. For the purpose of computing a transmission risk for case 3 we thus require the probabilities for each of the two scenarios (and for the pre-symptomatic scenario for each sub-scenario $T_S^A = t_0 + n$, for $n \geq 1$) which will then be appropriately weighted together. Thus, let

$$w_n = P(\mathcal{S}ymp_A(T_S^A = t_0 + n) \mid \mathcal{N}symp_A(T_S^A > t_0))$$

denote the probabilities for the different sub-cases of the pre-symptomatic scenario, i.e. w_n is the conditional probability for the event that A develops symptoms n days after the upload date, where $n \in \{1, \dots, N\}$ with N being some plausible maximum time delay between upload date and DSO. Analogously, we define

$$w_{\texttt{asymptomatic}} = P(\mathcal{A}symp_A \mid \mathcal{N}symp_A(T_S^A > t_0))$$

which denotes the probability for the completely asymptomatic scenario. Thus, for the pre-symptomatic situation, i.e. $1 \leq n \leq N$, we obtain

$$
\begin{aligned}
w_n &= P\Big(\mathcal{S}ymp_A(T_S^A = t_0 + n) \,\Big|\, \mathcal{S}ymp_A(T_S^A > t_0) \,\cup\, \mathcal{A}symp_A\Big) \\[2mm]
&= \frac{P\Big(\mathcal{S}ymp_A(T_S^A = t_0 + n) \,\cap\, (\mathcal{S}ymp_A(T_S^A > t_0) \,\cup\, \mathcal{A}symp_A)\Big)}{P\Big(\mathcal{S}ymp_A(T_S^A > t_0) \,\cup\, \mathcal{A}symp_A\Big)} \\[2mm]
&= \frac{P\Big(\mathcal{S}ymp_A(T_S^A = t_0 + n)\Big)}{P(\mathcal{S}ymp_A(T_S^A > t_0)) + P(\mathcal{A}symp_A)} \\[2mm]
&= \frac{P\Big(\mathcal{S}ymp_A(T_S^A = t_0 + n) \,\Big|\, \mathcal{S}ymp_A\Big) P(\mathcal{S}ymp_A)}{P\Big(\mathcal{S}ymp_A(T_S^A > t_0) \,\Big|\, \mathcal{S}ymp_A\Big) P(\mathcal{S}ymp_A) + P(\mathcal{A}symp_A)}
\end{aligned}
$$

In order to compute w_n we thus require the probability that symptom onset occurs n days after the upload date given that onset of symptoms will happen, i.e. $P(\mathcal{S}ymp_A(T_S^A = t_0 + n) \mid \mathcal{S}ymp_A)$. This is covered by the delay distribution `d_upload2symp` introduced further above. Furthermore we need the probability for any infected person to develop symptoms at all, i.e. $P(\mathcal{S}ymp_A)$. This probability was in Mizumoto et al. (2020) estimated to be $\hat{P}(\mathcal{S}ymp_A) = 0.86$. For $P(\mathcal{S}ymp_A(T_S^A > t_0) \mid \mathcal{S}ymp_A)$ we rely on the random time from data upload to symptom onset for suspected cases, who were tested in a pre-symptomatic stage.

Since according to `d_upload2symp` the maximum delay between upload and DSO is three days we set $N = 3$. By plugging in the individual numbers we can compute the scenario probabilities w_n for $n \in \{1, 2, 3\}$. For instance, for w_1 we obtain

$$w_1 = \frac{0.1383 \cdot 0.86}{(0.1383 + 0.0317 + 0.0033) \cdot 0.86 + 0.14} = 0.412$$

The overall results for w_n, $n = 1, 2, 3$, are given below:

```r
w_n <- (d_upload2symp[c("1", "2", "3")] * p_symp) /
  (sum(d_upload2symp[c("1", "2", "3")]) * p_symp + (1 - p_symp))

data.frame(n = 1:3, w_n = w_n)
```

```
##   n          w_n
## 1 1 0.411554428
## 2 2 0.094211255
## 3 3 0.009916974
```

Similarly, we obtain

$$
\begin{aligned}
w_{\texttt{asymptomatic}} &= P\Big(\mathcal{A}symp_A \,\Big|\, \mathcal{S}ymp_A(T_S^A > t_0) \,\cup\, \mathcal{A}symp_A\Big) \\
&= \frac{P(\mathcal{A}symp_A)}{P\Big(\mathcal{S}ymp_A(T_S^A > t_0)\,\Big|\,\mathcal{S}ymp_A\Big)P(\mathcal{S}ymp_A) + P(\mathcal{A}symp_A)} \\
&= \frac{0.14}{(0.1383 + 0.0317 + 0.0033)\cdot 0.86 + 0.14} = 0.484
\end{aligned}
$$

3.3.3.1   Computation of the Transmission Risk   Knowing the probabilities of the possible scenarios and sub-scenarios in the case of a person being non-symptomatic at the time of uploading a positive test result, we can compute the transmission risk for case 3 by applying the law of total probability, i.e.

$$
\begin{aligned}
&\lambda_A^{(raw)}(d, \mathcal{N}symp_A(T_S^A > t_0)) \\
&= \mathbb{E}\Big[\lambda_A^{(raw)}(d, \cdot) \,\Big|\, \mathcal{S}ymp_A(T_S^A > t_0) \,\cup\, \mathcal{A}symp_A\Big] \\
&= w_{\texttt{asymptomatic}} \cdot \lambda_A^{(raw)}(d, \mathcal{A}symp_A) + \sum_{n=1}^{3} w_n \cdot \lambda_A^{(raw)}(d, \mathcal{S}ymp_A(T_S^A = t_0 + n))
\end{aligned}
$$

Thus, the first term shows that we require the transmission risk function for completely asymptomatic people. Here we assume, that completely asymptomatic cases have the same infectiousness profile (as a function around the date of upload) like symptomatic people, but with their infectiousness reduced by a factor of 0.4 (see Mizumoto et al. 2020). However, note that in this case we cannot rely on the computations from case 2 (symptomatic with unknown DSO), since here the DSO does not follow the distribution `d_symp2upload`, which applies for people who were known to be symptomatic at the time of upload. For this case we instead focus on people who were tested in an non-symptomatic stage, such that the DSO (for people who develop symptoms) is distributed according to `d_upload2symp` subject to the upload date t_0. Utilizing the transmission risk from case 1 with known DSO, this yields

$$
\begin{aligned}
&\lambda_A^{(raw)}(d, \mathcal{A}symp_A) \\
&= 0.4 \cdot \mathbb{E}_{T_S^A}\Big[\lambda_A^{(raw)}(d, \mathcal{S}ymp_A(T_S^A = t_S^A)) \,\Big|\, \mathcal{S}ymp_A(T_S^A > T_{\text{sampling}}^A),\ \mathcal{S}ymp_A\Big] \\
&= 0.4 \cdot \sum_{t_S^A = t_0 - 3}^{t_0 + 3} \lambda_A^{(raw)}(d, \mathcal{S}ymp_A(T_S^A = t_S^A)) \cdot d_{\texttt{upload2symp}(t_S^A - t_0)}
\end{aligned}
$$

In code:

```r
### add days_since_symptoms from -3 to -1 to M_case1
M_case1_extended <- rbind(
  matrix(0,
    ncol = ncol(M_case1),
    nrow = 3,
```

```
      dimnames = modifyList(dimnames(M_case1), list(days_since_symptoms = seq(-3, -1)))
  ),
  M_case1
)

# Go backwards from delay 0 to -3 and shift \lambda_A by one to the left
for (rn in 3:1) {
  M_case1_extended[rn, ] <- c(M_case1_extended[rn + 1, -1], 0)
}

### case pre-symptomatic
M_case3_presymtomatic <- w_n %*% M_case1_extended[c("-1", "-2", "-3"), ]

### case non-symptomatic
M_case3_nonsymptomatic <- matrix(0,
  ncol = ncol(M_case1), nrow = 1,
  dimnames = modifyList(dimnames(M_case1), list(days_since_symptoms = NA_character_))
)

for (d in names(d_upload2symp)) {
  M_case3_nonsymptomatic <- M_case3_nonsymptomatic +
    (factor_asymp_reduce_infectiousness * M_case1_extended[d, ] * d_upload2symp[d])
}

### combined case
M_case3 <- M_case3_presymtomatic + w_asymptomatic * M_case3_nonsymptomatic

# Convert to data.frame
df_M_case3 <- matrix_to_df(M_case3)

# Show the scale
plot_rel_infectiousness(df_M_case3)
```
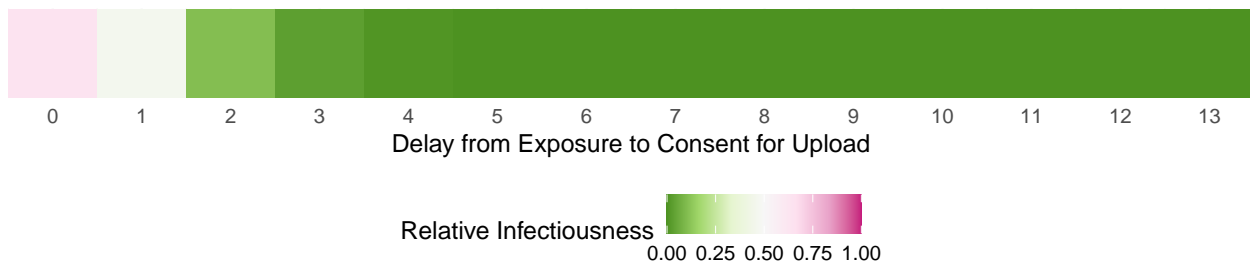

Figure 8: Raw transmission risk for Case 3: A is non-symptomatic at day of upload.

If person *A* has not developed any symptoms up to the day of upload, the transmission risk is highest at the day of upload itself, and gradually decays for each day that a risk contact lays further in the past.

3.3.4 Case 4: Unknown Symptom Status

The final case represents the scenario in which no further information is available beyond the fact that person *A* uploaded a positive test results at time t_0. This case covers several possible scenarios from above and summarizes them into a condensed transmission risk.

We decompose the case into the following possible mutually exclusive scenarios and apply corresponding probabilities for each subscenario:

(i) $\mathcal{S}ymp_A(T_S^A \leq T_{\text{sampling}}^A)$, i.e. person A got tested because they developed COVID-19 related symptoms,

(ii) $\mathcal{N}symp_A(T_S^A > T_{\text{sampling}}^A)$, i.e. person A got tested as a suspected case, e.g. because of a risk contact, and was non-symptomatic at the time of sampling. This can be decomposed into:

    (a) $\mathcal{S}ymp_A(T_S^A > T_{\text{sampling}}^A)$, i.e. person A develops symptoms after being tested,

    (b) $\mathcal{A}symp_A \cap \mathcal{N}symp_A(T_S^A > T_{\text{sampling}}^A)$, i.e. person A is a completely asymptomatic case and will thus never develop symptoms.

In summary we get the following disjoint decomposition:

$$\Omega = \mathcal{S}ymp_A(T_S^A \leq T_{\text{sampling}}^A) \ \cup \ \mathcal{S}ymp_A(T_S^A > T_{\text{sampling}}^A) \ \cup \ \left(\mathcal{A}symp_A \cap \mathcal{N}symp_A(T_S^A > T_{\text{sampling}}^A) \right).$$

The transmission risk as a function of time difference d between day of upload and day of contact for each of these scenarios can be derived based on results and considerations from the above cases 1 to 3.

For (i) we can apply the risk value function from case 2 (symptomatic with unknown DSO), i.e.

$$\lambda_A^{(raw)}(d, \mathcal{S}ymp_A(T_S^A \leq T_{\text{sampling}}^A)) = \lambda_A^{(raw)}(d, \mathcal{S}ymp_A(T_S^A \leq t_0)).$$

For (ii.a) we can rely on the same considerations as made for case 3 (non-symptomatic at time of upload). Again, note that for the here reflected scenario we assume that person A was pre-symptomatic at time of testing. For considering the potential DSOs, we therefore again rely on the delay distribution between upload and DSO used for pre-symptomatically tested people, i.e. `d_upload2symp`. Thus, we compute the expected risk value by marginalizing over the DSO which follows `d_upload2symp` subject to the given date of upload t_0, i.e.

$$\lambda_A^{(raw)}(d, \mathcal{S}ymp_A(T_S^A > T_{\text{sampling}}^A))$$
$$= \mathbb{E}_{T_S^A} \left[\lambda_A^{(raw)}(d, \mathcal{S}ymp_A(T_S^A = t_S^A)) \,\Big|\, \mathcal{S}ymp_A(T_S^A > T_{\text{sampling}}^A) \right]$$
$$= \sum_{t_S^A = t_0 - 3}^{t_0 + 3} \lambda_A^{(raw)}(d, \mathcal{S}ymp_A(T_S^A = t_S^A)) \cdot d_{\texttt{upload2symp}(t_S^A - t_0)}$$

For (ii.b) we follow the same arguments as in case 3 for completely asymptomatic cases, which effectively yields $\lambda_A^{(raw)}(d, \mathcal{S}ymp_A(T_S^A > T_{\text{sampling}}^A))$ times a factor for the reduced relative infectiousness, i.e.

$$\lambda_A^{(raw)}(d, \mathcal{A}symp_A \cap \mathcal{N}symp_A(T_S^A > T_{\text{sampling}}^A))$$
$$= \lambda_A^{(raw)}(d, \mathcal{A}symp_A)$$
$$= 0.4 \cdot \lambda_A^{(raw)}(d, \mathcal{S}ymp_A(T_S^A > T_{\text{sampling}}^A))$$

To define the overall transmission risk for case 4, we require probabilities for how likely each of the distinct scenarios is to occur. In order to weigh scenarios (ii.a) and (ii.b) we again rely on the probability that a proportion of 86% of infected cases develop symptoms (Mizumoto et al. (2020)). This leaves the question on what proportion of cases, who upload their positive test result, were already tested while being asymptomatic. For now, we can only assume that this proportion is rather low, at $p_{\texttt{suspect}} = 0.2$. However, the proportion $p_{\texttt{suspect}}$ is likely subject to change over time, in particular due to the impact of the *Corona-Warn-App* itself, as it has the primary purpose of identifying potentially infected people and initiate corresponding tests in an early stage of their course of infection. Altogether, this leads to the transmission risk for case 4 being defined as follows.

$$\lambda_A^{(raw)}(d, \Omega)$$
$$= (1 - p_{\text{suspect}}) \cdot \lambda_A^{(raw)}(d, \mathcal{S}ymp_A(T_S^A \le T_{\text{sampling}}^A)) +$$
$$p_{\text{suspect}} \cdot 0.86 \cdot \lambda_A^{(raw)}(d, \mathcal{S}ymp_A(T_S^A > T_{\text{sampling}}^A)) +$$
$$p_{\text{suspect}} \cdot (1 - 0.86) \cdot \lambda_A^{(raw)}(d, \mathcal{A}symp_A \cap \mathcal{N}symp_A(T_S^A > T_{\text{sampling}}^A))$$

This leads to the following transmission risks.

```
### combined case
M_case4 <- (1 - p_suspect) * M_case2 +
  p_suspect * p_symp * (1 / factor_asymp_reduce_infectiousness) * M_case3_nonsymptomatic +
  p_suspect * (1 - p_symp) * M_case3_nonsymptomatic


# Convert result to matrix for better comparison
M_case4 <- matrix(M_case4,
  ncol = max_dse + 1,
  nrow = 1,
  dimnames = list("no information at upload", days_since_contact = 0:max_dse)
)


# Convert to data.frame
df_M_case4 <- matrix_to_df(M_case4)


# Show the scale
plot_rel_infectiousness(df_M_case4)
```


Figure 9: Raw transmission risk for Case 4: no information about A at upload.

Note that case 4 represents the baseline case that is applied if the app does not or cannot collect any additional information about person A, which will apply at least for the initial public version of the *Corona-Warn-App*.

3.4 Transmission Risk Level

The final step for generating the *transmission risk level* is to translate the raw transmission risk from the continuous scale $[0, 1]$ into a discrete scale (categories from *I* to *VIII* in roman numbers).

To do so, we set equidistant thresholds within the $[0, 1]$-scale, such that the scale is decomposed into 8 equal-sized intervals. A raw transmission risk $\lambda_A^{(raw)}(d, \cdot)$ within such an interval is then mapped onto the respective level $\lambda_A(d, \cdot)$, where the $\cdot$ stands for one of the four case scenarios, i.e.

$$\lambda_A(d, \cdot) = l \quad \Leftrightarrow \quad \frac{l-1}{8} \le \lambda_A^{(raw)}(d, \cdot) < \frac{l}{8},$$

where $l \in \{1, ..., 8\}$. Note: we define that raw values with $\lambda_A^{(raw)}(d, \cdot) = 1$ are also classified as *transmission risk level* VIII.

3.4.1 Transmission Risk Levels for Cases 1–4

This classification rule leads to the following risk levels for the raw risks obtained from each of the above considered case scenarios.
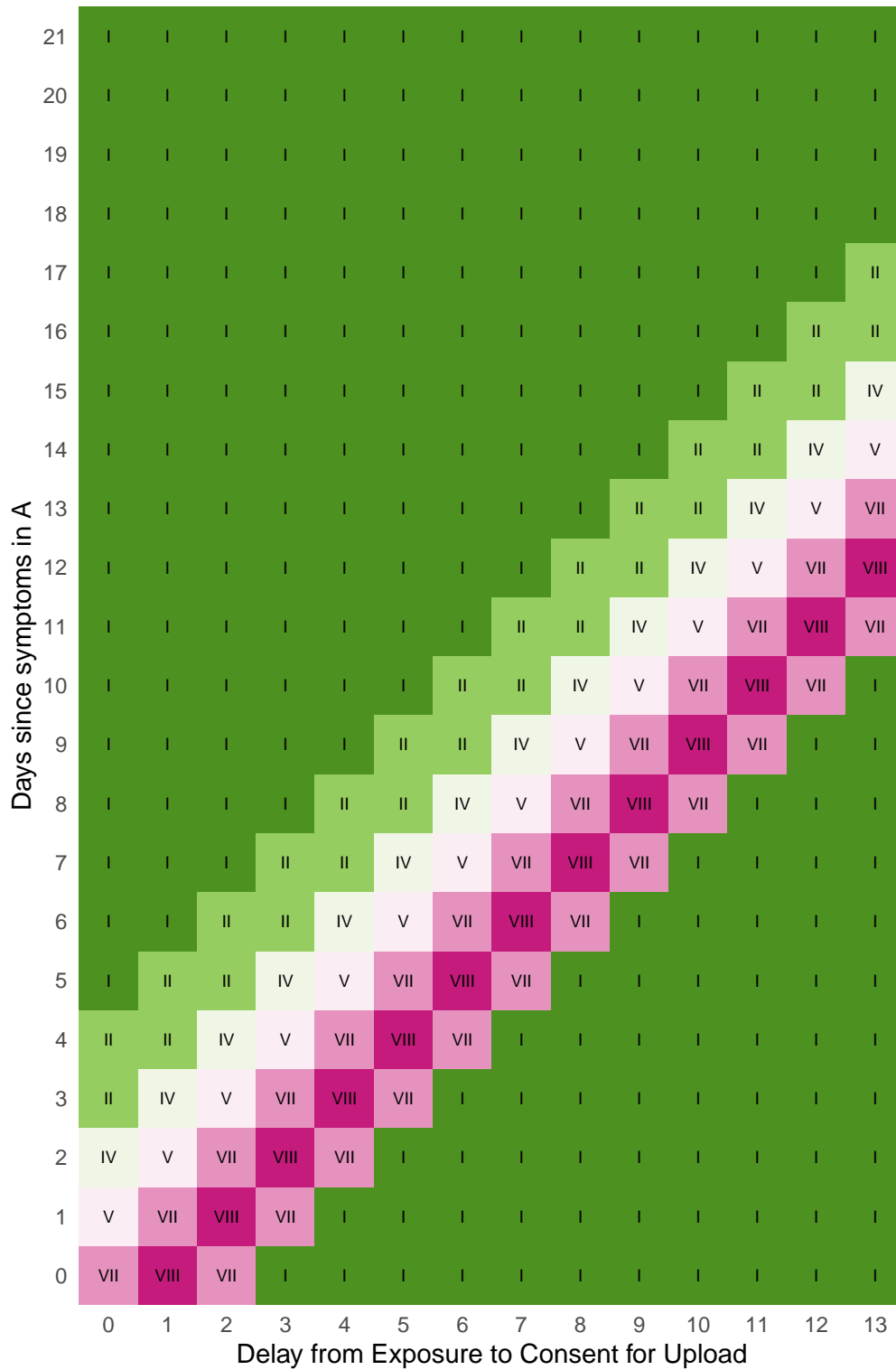

Figure 10: Transmission risk level for Case 1: A is symptomatic with DSO provided.
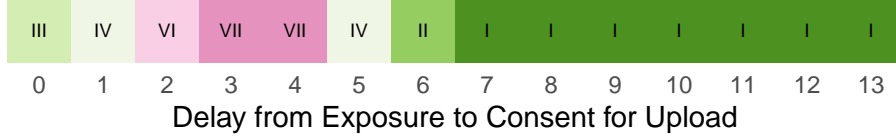
Figure 11: Transmission risk level for Case 2: A is symptomatic but with unknown DSO.
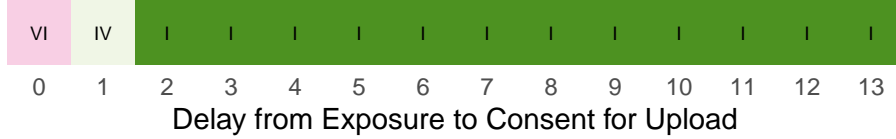

Figure 12: Transmission risk level for Case 3: A is non-symptomatic at day of upload.
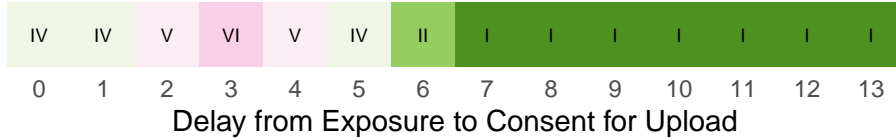

Figure 13: Transmission risk level for Case 4: no information about A at upload.

3.4.2 Transmission Risk Levels for the Base Case

The initial version of the *Corona-Warn-App* does not offer the possibility to provide additional information on symptom status when uploading a positive test result. Thus, the transmission risk is always calculated as in case 4, where no further information is available, i.e. the event Ω.

For the risk level calculation, the transmission risks are classified into levels based on the highest achievable value from case 1, in which the DSO is known. This leads to the highest possible *transmission risk level* in case 4 being at VI. However, since no additional information is provided for the initial version of the app, this case does never actually apply. Due to averaging over all possible scenarios in case 4, selecting the transmission level from case 4 in the above calculations implies that the *transmission risk level* has an upper level of VI. Therefore the app does not make use of the full level scale for the transmission risk, in particular levels VII and VIII are effectively never used. This could mean that the transmission risk weights less in the subsequent *total risk score* computations.

To address this issue we re-adjust the level classification by accounting for the maximum possible transmission risk achieved within case 4. For this purpose we will also refer to case 4 as the *base case* since it represents the only possible case within the initial app version. Thus we obtain

$$\lambda_A(d) = \lambda_A(d, \text{Base case}) = l \quad \Leftrightarrow \quad \frac{l-1}{8} \leq \frac{\lambda_A^{(raw)}(d, \Omega)}{\max_{d \in \{0\ldots,13\}} \lambda_A^{(raw)}(d, \Omega)} < \frac{l}{8},$$

where $l \in \{1, \ldots, 8\}$ and where again raw values with $\lambda_A^{(raw)}(d, \Omega) = \max_{d \in \{0\ldots,13\}} \lambda_A^{(raw)}(d, \Omega)$ are also classified as level 8. This leads to the following risk levels for this single base case:

Figure 14: Transmission risk level for the Base Case.

18

These are the *transmission risk levels* to be used for the initial app version. Once the app offers the possibility to provide additional information, which then enables the cases 1 to 3, we advise to fall back to the case-specific risk levels $\lambda_A(d, \cdot)$ as defined above (and maybe adjust alarm thresholds within the app accordingly).

4 Discussion

The present document reflects an epidemiological motivation of the *transmission risk level* used as part of the Corona-Warn-App. In particular we show how to inform this parameter by epidemiological information about COVID-19 and the processes leading to a confirmed diagnosis in order to identify periods of increased infectiousness. A stochastic framework was used to characterize the sequence of events. However, any such stochastic model is subject to assumptions and parameter choices, which need to be carefully evaluated. Given the present state of information about COVID-19 and the current simultaneous demands on resources, such an extensive evaluation and sensitivity analysis was not possible. As a consequence the present document reflects our best knowledge at the time of publication, but we strongly urge that relevant areas are identified for further evaluation. In particular this could consist in quantifying some of the operational delays from real-world data, once the app is in use.

With regards to some of the other parameters, a special challenge lies in the decentralized structure of the app. This was a non-negotiable design decision in the run-up of developing the app. However, in order to improve the app, epidemiological studies with a sample of voluntary users could be designed, in order to evaluate the appropriateness of not just the *transmission risk level*, but the overall classificational approach of the app.

One component to evaluate continuously should be the infectiousness profile utilized in this model. Our current estimate inspired by He et al. (2020) contains several time- and context-specific quantities such as the contact structure and the resulting serial interval. As an example, the serial interval is a time-dependent quantity of an outbreak, e.g., because susceptible persons are depleted (Svensson (2007)), or because increased contact tracing efforts, reduce the period of infectiousness. Finally, the introduction and increased and successful use of the app may alter the very parameters that we estimated for its calibration.

Altogether, it would be important to continuously evaluate and update the infectiousness profile, e.g., by being better able to connect information about viral shedding with the risk of infection. Another important limitation is that throughout the document we assumed that *A* infected *B*. However, the likelihood and the consequences that it was *B* who infected *A* should be further investigated - this seems particularly important in a setting with many asymptomatic or pre-symptomatic transmissions.

One important point of this document is, that a better transmission risk assessment would be possible, if information about day of symptom onset of the person uploading his or her *diagnosis keys* would be available. The specific gain, for example in terms of sensitivity and specificity, is currently not provided, but could be quantified using simulation.

However, besides the epidemiological aspect two other aspects are important: Firstly, using additional information about *A* and mapping that to levels from I-VIII can be a privacy risk, because a potential attacker could reveal some information about *A* from level and time of upload. In the specific case, if provided, the DSO of *A* can probably be inferred in some situations. Secondly, the information about DSO will be self-reported. In our computations we assume DSO to be reported correctly, but misreports are likely to occur, which could also lead to serious bias in the *transmission risk level* A malicious user can also falsely report his or her DSO to cause false alarms. Hence, one has to carefully balance these two aspects, e.g., by evaluating different formats of DSO specification in the App. For a thoroughly discussion of potential risk of a proximity tracing app see Privacy and Security Attacks on Digital Proximity Tracing Systems.

The epidemiological motivated *transmission risk level* has value in contact tracing beyond the App. Close work together with local health authorities and those actually performing the contact tracing in the field is, however, necessary to make this added-value more specific.

4.1 Quality Assurance

The calculations in the present document reflect one of two implementations, which were made as part of a two-mindset approach for quality assurance of the *transmission risk level* computations.

4.2 Acknowledgements

We thank Eric H. Y. Lau for his quick and friendly response to some additional questions regarding underlying data of the He et al. (2020) publication.

5 Appendix

5.1 Convolution of Discrete Random Variables

In what follows we will use the fact that for two *independent* integer random variables X and Y with PMFs f_X and f_Y, respectively, we will have for their sum $Z = X + Y$ that

$$f_Z(z) = \sum_x f_X(x) f_Y(z - x).$$

This can be handled in R using the following function, where we assume that the PMF is formulated as a vector where the elements denote the individual probabilities and the names of the vector contain the support of the distribution.

```r
#' Function to convolute two discrete probability distributions with
#' support on 0, 1, ... I.e. we compute the PMF of Z = X + Y.
#'
#' @param fX - the PMF of X given as a named vector, where the names
#'             represent the support X_min, ..., 0, 1, ..., X_max
#' @param fY - the PMF of Y given as a named vector, where the names
#'             represent the support Y_min, ..., 0, 1, ..., Y_max
#'
#' @return A names vector containing the PMF of Z = X+Y.
#'
#' @examples
#' convolute(
#'   c(`-2` = .2, `-1` = .3, `0` = .2, `1` = .1, `2` = .1),
#'   c(`-1` = .2, `0` = .3, `1` = .2, `2` = .2, `3` = 0.1)
#' )
convolute <- function(fX, fY) {
  fZ <- rep(0, 1 + (length(fX) - 1) + (length(fY) - 1))
  names(fZ) <- as.character(seq(
    min(as.numeric(names(fX))) + min(as.numeric(names(fY))),
    max(as.numeric(names(fX))) + max(as.numeric(names(fY)))
  ))

  for (i in names(fX)) {
    for (k in names(fY)) {
      j <- as.numeric(i) + as.numeric(k)
      fZ[as.character(j)] <- fZ[as.character(j)] + fX[i] * fY[k]
    }
  }

  fZ
}
```

5.2 Helper Functions for Discrete PMFs and Plots

```r
#' Helper function to convert a names vector containing a PMF to a data.frame
#'
#' @param pmf - the PMF of X given as a named vector, where the names
#'              represent the support X_min, ..., 0, 1, ..., X_max
#'
#' @return A data frame containing the PMF with columns for name (x) and value (pmf).
#'
#' @examples
#' pmf2df(c("-1" = 0.2, "0" = 0.3, "1" = 0.4, "2" = 0.1))
pmf2df <- function(pmf) {
  data.frame(x = as.numeric(names(pmf)), pmf = pmf)
}


#' Helper function to draw a PMF
#'
#' @examples
#' ggplot_pmf(c("-1" = 0.2, "0" = 0.3, "1" = 0.4, "2" = 0.1))
ggplot_pmf <- function(pmf) {
  ggplot(pmf2df(pmf), aes(x = x, ymin = 0, ymax = pmf)) +
    geom_linerange() +
    ylab("PMF")
}


#' Helper function to plot the risk levels
plot_risk_levels <- function(data, title = "",
                             breaks_y = NULL, ylab_text = "",
                             show_roman = TRUE,
                             fill_limits = c(1, 8),
                             fill_name = NULL) {
  ggplot(data, aes(x = y, y = x)) +
    geom_tile(aes(fill = M)) +
    geom_text(aes(label = as.character(M_roman)), size = 2.5) +
    xlab("Delay from Exposure to Consent for Upload") +
    ylab(ylab_text) +
    #labs(title = title) +
    scale_fill_distiller(
      palette = "PiYG", limits = fill_limits
    ) +
    scale_y_continuous(breaks = breaks_y) +
    scale_x_continuous(breaks = 0:max_dse) +
    theme_minimal() +
    coord_fixed(expand = FALSE) +
    guides(fill = FALSE)
}


#' Helper function to plot the relative infectiousness
plot_rel_infectiousness <- function(data, breaks_y = NULL, ylab_text = "") {
  ggplot(data, aes(x = y, y = x)) +
    geom_tile(aes(fill = M)) +
    xlab("Delay from Exposure to Consent for Upload") +
    ylab(ylab_text) +
    scale_fill_distiller(
```

```
      name = "Relative Infectiousness",
      palette = "PiYG", limits = c(0, 1)
    ) +
    scale_y_continuous(breaks = breaks_y) +
    scale_x_continuous(breaks = 0:max_dse) +
    theme_minimal() +
    coord_fixed(expand = FALSE) +
    theme(legend.position = "bottom")
}
```

5.3   Discretization of the Values in a Matrix

```
#' Take the values in the matrix mat and divide it into 8 equidistant levels
#' The result is returned as a df in long format with rows and cols variables
discretize_matrix <- function(mat, max_value = max(mat, na.rm=TRUE)) {
  cuts <- cut(
    mat,
    breaks = seq(0, max_value, length = 9),
    right = FALSE,
    include.lowest = TRUE
  )
  mat <- matrix(
    as.numeric(cuts),
    nrow = nrow(mat),
    ncol = ncol(mat),
    byrow = FALSE
  )
  data.frame(
    x = as.vector(row(mat) - 1),
    y = as.vector(col(mat) - 1),
    M = as.vector(mat),
    M_roman = as.character(as.roman(as.vector(mat))),
    stringsAsFactors = TRUE
  )
}
```

Literature

Apple, and Google. 2020a. "Exposure Notifications Android Api Documentation V1.3.2." 2020. https://st atic.googleusercontent.com/media/www.google.com/en/covid19/exposurenotifications/pdfs/Android-Exposure-Notification-API-documentation-v1.3.2.pdf.

———. 2020b. "Exposure Notifications Cryptography Specification V1.2." 2020. https://covid19-stat ic.cdn-apple.com/applications/covid19/current/static/contact-tracing/pdf/ExposureNotification-CryptographySpecificationv1.2.pdf.

Arons, Melissa M., Kelly M. Hatfield, Sujan C. Reddy, Anne Kimball, Allison James, Jesica R. Jacobs, Joanne Taylor, et al. 2020. "Presymptomatic Sars-Cov-2 Infections and Transmission in a Skilled Nursing Facility." *New England Journal of Medicine* 382 (22): 2081–90. https://doi.org/10.1056/NEJMoa200845 7.

He, Xi, Eric H. Y. Lau, Peng Wu, Xilong Deng, Jian Wang, Xinxin Hao, Yiu Chung Lau, et al. 2020. "Temporal Dynamics in Viral Shedding and Transmissibility of Covid-19." *Nature Medicine* 26 (5): 672–75. https://doi.org/10.1038/s41591-020-0869-5.

Mizumoto, Kenji, Katsushi Kagaya, Alexander Zarebski, and Gerardo Chowell. 2020. "Estimating the Asymptomatic Proportion of Coronavirus Disease 2019 (Covid-19) Cases on Board the Diamond Princess Cruise Ship, Yokohama, Japan, 2020." *Eurosurveillance* 25 (10). https://doi.org/https://doi.org/10.2807/1560-7917.ES.2020.25.10.2000180.

Svensson, Å. 2007. "A note on generation times in epidemic models." *Math Biosci* 208 (1): 300–311. https://doi.org/10.1016/j.mbs.2006.10.010.

Wölfel, Roman, Victor M. Corman, Wolfgang Guggemos, Michael Seilmaier, Sabine Zange, Marcel A. Müller, Daniela Niemeyer, et al. 2020. "Virological Assessment of Hospitalized Patients with Covid-2019." *Nature* 581 (7809): 465–69. https://doi.org/10.1038/s41586-020-2196-x.