Interdisciplinary Sciences: Computational Life Sciences
https://doi.org/10.1007/s12539-021-00439-2

ORIGINAL RESEARCH ARTICLE

# On the Identification of Clinically Relevant Bacterial Amino Acid Changes at the Whole Genome Level Using Auto-PSS-Genome

**Hugo López-Fernández[1,2,3,4,5] · Cristina P. Vieira[4,5] · Pedro Ferreira[4,5] · Paula Gouveia[4,5] · Florentino Fdez-Riverola[1,2,3] · Miguel Reboiro-Jato[1,2,3] · Jorge Vieira[4,5]**

Received: 4 December 2020 / Revised: 21 March 2021 / Accepted: 7 May 2021


## Abstract

The identification of clinically relevant bacterial amino acid changes can be performed using different methods aimed at the identification of genes showing positively selected amino acid sites (PSS). Nevertheless, such analyses are time consuming, and the frequency of genes showing evidence for PSS can be low. Therefore, the development of a pipeline that allows the quick and efficient identification of the set of genes that show PSS is of interest. Here, we present Auto-PSS-Genome, a Compi-based pipeline distributed as a Docker image, that automates the process of identifying genes that show PSS using three different methods, namely codeML, FUBAR, and omegaMap. Auto-PSS-Genome accepts as input a set of FASTA files, one per genome, containing all coding sequences, thus minimizing the work needed to conduct positively selected sites analyses. The Auto-PSS-Genome pipeline identifies orthologous gene sets and corrects for multiple possible problems in input FASTA files that may prevent the automated identification of genes showing PSS. A FASTA file containing all coding sequences can also be given as an external global reference, thus easing the comparison of results across species, when gene names are different. In this work, we use Auto-PSS-Genome to analyse *Mycobacterium leprae* (that causes leprosy), and the closely related species *M. haemophilum,* that mainly causes ulcerating skin infections and arthritis in persons who are severely immunocompromised, and in children causes cervical and perihilar lymphadenitis. The genes identified in these two species as showing PSS may be those that are partially responsible for virulence and resistance to drugs.

✉ Jorge Vieira
jbvieira@ibmc.up.pt

Hugo López-Fernández
hlfernandez@uvigo.es

Cristina P. Vieira
cgvieira@ibmc.up.pt

Pedro Ferreira
pedro.ferreira@i3s.up.pt

Paula Gouveia
paula.gouveia.mdl@gmail.com

Florentino Fdez-Riverola
riverola@uvigo.es

Miguel Reboiro-Jato
mrjato@uvigo.es

[1] Department of Computer Science, University of Vigo, ESEI, Campus As Lagoas, 32004 Ourense, Spain

[2] The Biomedical Research Centre (CINBIO), Campus Universitario Lagoas-Marcosende, 36310 Vigo, Spain

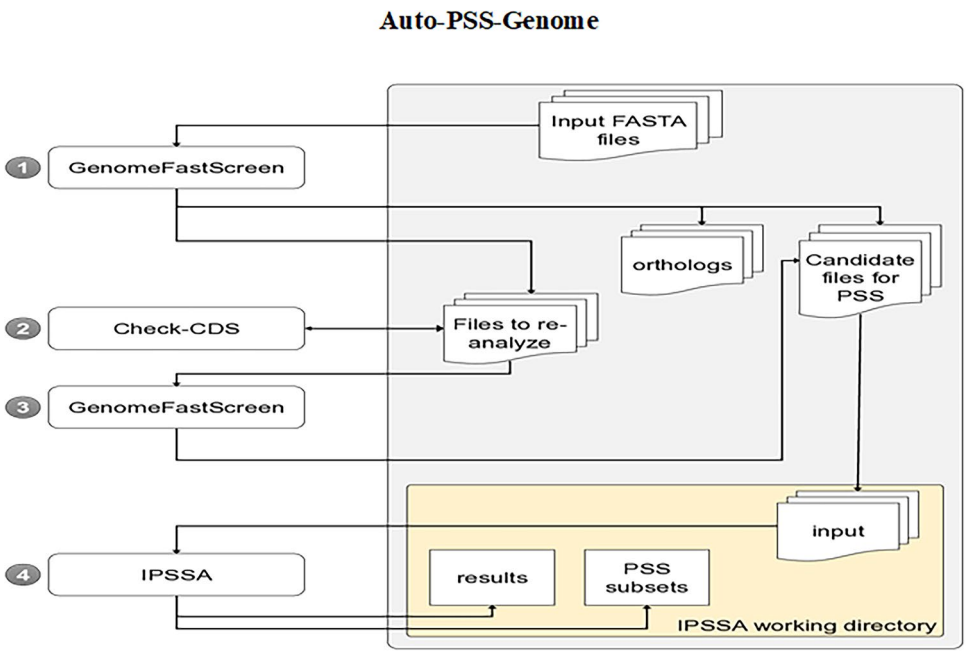
[3] SING Research Group, Galicia Sur Health Research Institute (IIS Galicia Sur), SERGAS-UVIGO, Vigo, Spain

[4] Instituto de Investigação e Inovação em Saúde (I3S), Universidade do Porto, Rua Alfredo Allen, 208, 4200-135 Porto, Portugal

[5] Instituto de Biologia Molecular e Celular (IBMC), Rua Alfredo Allen, 208, 4200-135 Porto, Portugal

Published online: 19 May 2021

**Graphic Abstract**




**Keywords** Bacteria · Positively selected amino acid sites · Big data · Compi

## 1 Introduction

Adaptive features such as, but not limited to, host immune response escape or antibiotic resistance can be determined by protein amino acid changes. Amino acid positions that show more changes than expected under a neutral evolutionary model are under positive selection and are named positively selected amino acid sites (PSS). PSS can be inferred using sequence data and phylogenetic (codeML [1] and FUBAR [2], for instance), or population-based methods (omegaMap [3], for instance). Therefore, it is of interest to perform these analyses at the genome level, especially in the case of pathogens. Here, we present Auto-PSS-Genome, a flexible and fully automated pipeline that can be used to identify genes that show PSS using three different methods, namely codeML, FUBAR, and omegaMap, starting from at least four FASTA files, each containing the coding sequences (CDS) that were annotated on that particular genome. The FASTA files can be downloaded, for instance, from the NCBI RefSeq database, by querying for the species of interest at https://www.ncbi.nlm.nih.gov/assembly. The rationale behind the use of multiple methods is that, since both phylogenetic and population genetics methods have weaknesses, inferences on PSS should not rely on a single method.

Among the available methods that can be used to detect PSS, FUBAR is the one that requires the least amount of time to run. Nevertheless, since this method requires both an aligned set of CDS as well as a phylogenetic tree describing the relationship among them, large-scale detailed analyses can still take a large amount of time. Since only a small fraction of all genes will show PSS, the quick identification of the subset of genes that will likely show PSS when analysed in detail becomes necessary. Pipelines such as FastScreen [4] and GenomeFastScreen [5] (an automated version of FastScreen for analyses involving whole genomes) serve this purpose, allowing researchers to save a substantial amount of computational and research time, and thus were included in the Auto-PSS-Genome pipeline. For instance, Osório et al. [6] found evidence for PSS in only five out of 576 genes that were previously associated with drug resistance or encoding membrane proteins, when using 73 publicly available genomes from all the main *Mycobacterium tuberculosis* (the causative agent of tuberculosis) complex lineages. Some of the identified PSS correspond to the position of confirmed drug-resistance-associated substitutions in the genes *embB*, *rpoB*, and *katG* [6], thus supporting the robustness of this approach. Moreover, we have recently found evidence for PSS in 31 out of 1587 *M. leprae* (the causative agent of leprosy) genes that could be analysed, including nine that are likely clinically relevant in the context of leprosy [5]. In that work, the GenomeFastScreen pipeline was used to create a short list of genes that

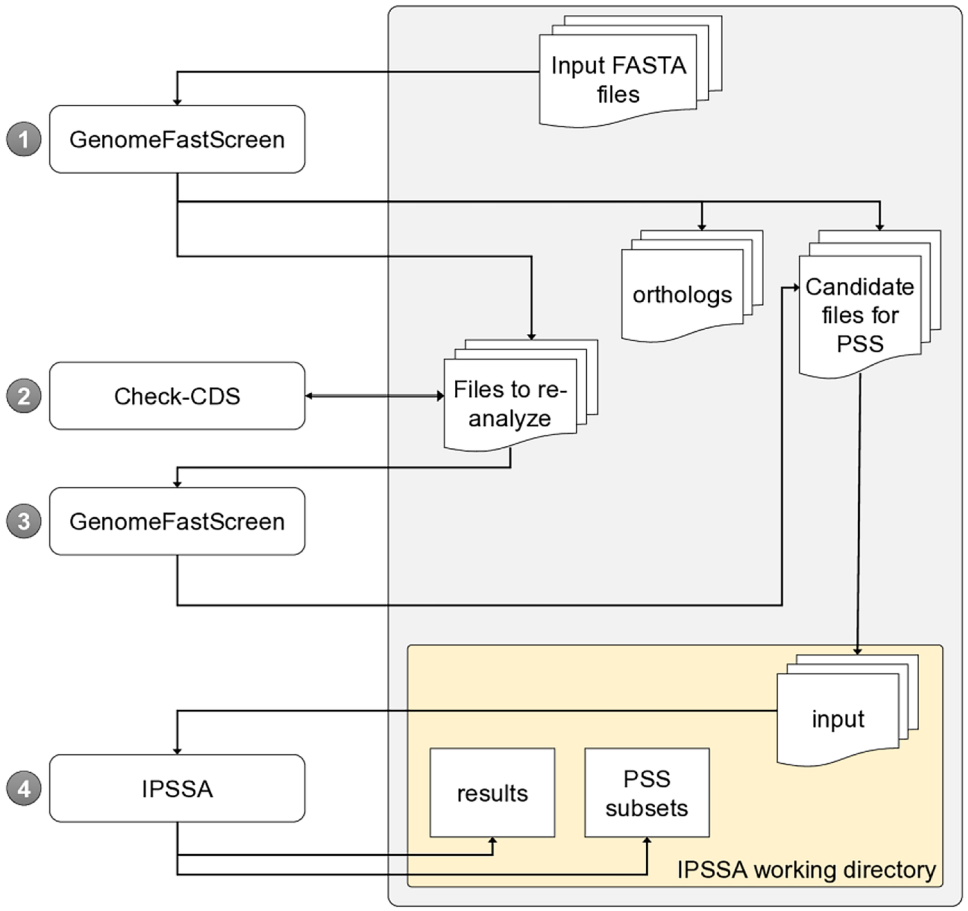



**Fig. 1** Steps and files involved in the Auto-PSS-Genome pipeline

when analysed in detail are likely to show PSS, and then, detailed analyses were performed using codeML models M1a, M2a, M7, and M8 using ADOPS [7].

To illustrate the usefulness of the Auto-PSS-Genome pipeline, we re-ran the analyses previously performed for *M. leprae* to see how robust the conclusions that were drawn are when three methods for detecting PSS are used (codeML, FUBAR, and omegaMap). In addition, we used Auto-PSS-Genome to analyse *M. haemophilum*, a species frequently found in environmental habitats, which can occasionally infect humans, causing ulcerating skin infections and arthritis in persons who are severely immunocompromised, and in healthy children leads to the development of cervical and perihilar lymphadenitis [8]. Finally, we compare the results obtained for *M. leprae* and *M. haemophilum*. This way we can find if the genes showing evidence for PSS are the same in two closely related species that cause very different diseases. It should be noted that *M. lepromatosis* is more closely related to *M. leprae* than *M. haemophilum* [9], but there is a single genome at the NCBI Assembly database for *M. lepromatosis*, and thus this species could not be used for this purpose. Therefore, *M. haemophilum*, the sister species to *M. leprae*/*M. lepromatosis* is here used.

## 2 Materials and Methods

### 2.1 The Auto-PSS-Genome Compi Pipeline

The Auto-PSS-Genome[1] pipeline here reported is implemented as a Compi pipeline and distributed as a Docker image that allows running it effortlessly. The source code of the pipeline is publicly available at GitHub[2] and the Docker image at Docker Hub.[3] The Auto-PSS-Genome pipeline relies on the usage of three Compi pipelines developed by us. On one hand, GenomeFastScreen Compi pipeline recently reported [5], that uses the FastScreen Compi pipeline [4]. On the other hand, two new Compi pipelines specifically developed for this work, namely CheckCDS and Integrated Positively Selected Sites Analyses (IPSSA). These two new developments are described in detail in Sects. 2.2 and 2.3. The Auto-PSS-Genome repositories provide detailed

[1] https://www.sing-group.org/compihub/explore/5faa52ccf05e940c9c2762e4.

[2] https://github.com/pegi3s/auto-pss-genome.

[3] https://hub.docker.com/r/pegi3s/auto-pss-genome.

instructions on how to run the pipeline with the sample data made available by us.

As Fig. 1 illustrates, Auto-PSS-Genome accepts as input the annotation of the genomes to be analysed as a FASTA formatted file, one per genome. First, it analyses these input files using GenomeFastScreen. This pipeline allows finding orthologous genes using a two-way BLAST approach, and performs several sanity checks of the input FASTA files. These checks include making sure that sequences are multiple of three (if not, the pipeline automatically corrects them using a reference protein sequence), removing sequences with in-frame stop codons and/or ambiguous positions, and removing stop codons at the end of CDS, if present. This way, we guarantee that the pipeline is able to analyze the input data even when some problems are present. The GenomeFastScreen pipeline produces a list of files that are more likely to reveal PSS when analysed in detail than the ones not included in the list. To be as fast as possible, the alignment is performed at the nucleotide level using Clustal Omega, that performs big alignments quickly and accurately [10], FastTree, that is two to three orders of magnitude faster than the PhyML 3.0 or RAxML 7 alternatives [11], is used for inferring a phylogeny, and PSS are inferred using FUBAR and only one (M2a) out of the six available codeML models. Since both FastScreen and GenomeFastScreen have been described previously, they will not be here further detailed. The following two subsections describe how the Auto-PSS-Genome pipeline uses the CheckCDS and IPSSA pipelines.

### 2.2 The CheckCDS Compi Pipeline

As reported in [5], for two out of 1597 genes analysed, the GenomeFastScreen Compi pipeline produced an error, because at least one sequence in these files presents non-multiple of three alignment gaps when compared to the other sequences in the same file, leading to a non-multiple of three-nucleotide alignment. It should be noted that this could only happen if at least one sequence is annotated wrongly in the corresponding genome, or if the gene is a pseudogene in at least one genome. Very likely, this number will increase as the number of annotated genomes to be used increases. GenomeFastScreen puts the files that show such errors in a folder named "files_requiring_attention", allowing researchers to fix them and re-run the analyses. These are the *Files to re-analyze* in Fig. 1.

The CheckCDS[4] Compi Docker image here reported uses a greedy approach to solve this problem. It starts with the sequence indicated by the user as the reference and adds another sequence from the same sequence dataset to perform a nucleotide sequence alignment using the fast alignment algorithm Clustal Omega [10]. Then, it checks whether the resulting sequence alignment is multiple of three. If this is the case, it adds another sequence, performs again the alignment step using all accepted sequences and checks whether the resulting sequence alignment is multiple of three. If not, the sequence is removed from the output file. These steps are repeated until every available sequence is used.

After using the CheckCDS method to remove the problematic sequences (step 2 in Fig. 1), the Auto-PSS-Genome can now run the FastScreen step of the GenomeFastScreen pipeline without producing an error (step 3 in Fig. 1).

The source code of CheckCDS is publicly available at GitHub[5] and the Docker image at Docker Hub.[6]

### 2.3 The Integrated Positively Selected Sites Analyses (IPSSA) Compi Pipeline

GenomeFastScreen generates a short list of genes that when analysed in detail are more likely to reveal PSS than those not included in it. These files are the *Candidate files for PSS* in Fig. 1. The Integrated Positively Selected Sites Analyses[7] (IPSSA) Compi pipeline here presented, allows the user to perform such detailed analyses using three available approaches and those *Candidate files* files as input (step 4 in Fig. 1). The available approaches include two phylogenetic methods, FUBAR [2], and codeML [1] (models M1a and M2a, M7, and M8 can be chosen), as well as one population-based method, omegaMap [3].

Figure 2 shows the main steps of the IPSSA pipeline. IPSSA accepts non-aligned CDS files in FASTA format. The first step removes stop codons and line breaks in each input file to guarantee that the subsequent steps can run without problems. The second step checks if there are ambiguous nucleotide positions or non-multiple of three sequences. If so, the pipeline execution is stopped and the user must correct such files (or remove them) before continuing. In the context of the Auto-PSS-Genome, this situation is not expected to occur, since the input files are produced by the GenomeFastScreen pipeline, but the IPSSA pipeline is also provided as an independent pipeline and thus it is important to check the validity of the input files before starting the analyses.

If the number of sequences in the dataset is larger than specified by the user, a random sample with the desired number of sequences is first obtained to produce the master

[4] http://sing-group.org/compihub/explore/5f588ccb407682001ad3a1d5.

[5] https://github.com/pegi3s/check-cds.

[6] https://hub.docker.com/r/pegi3s/check-cds.

[7] https://sing-group.org/compihub/explore/5fa91806407682001ad3a1e9.

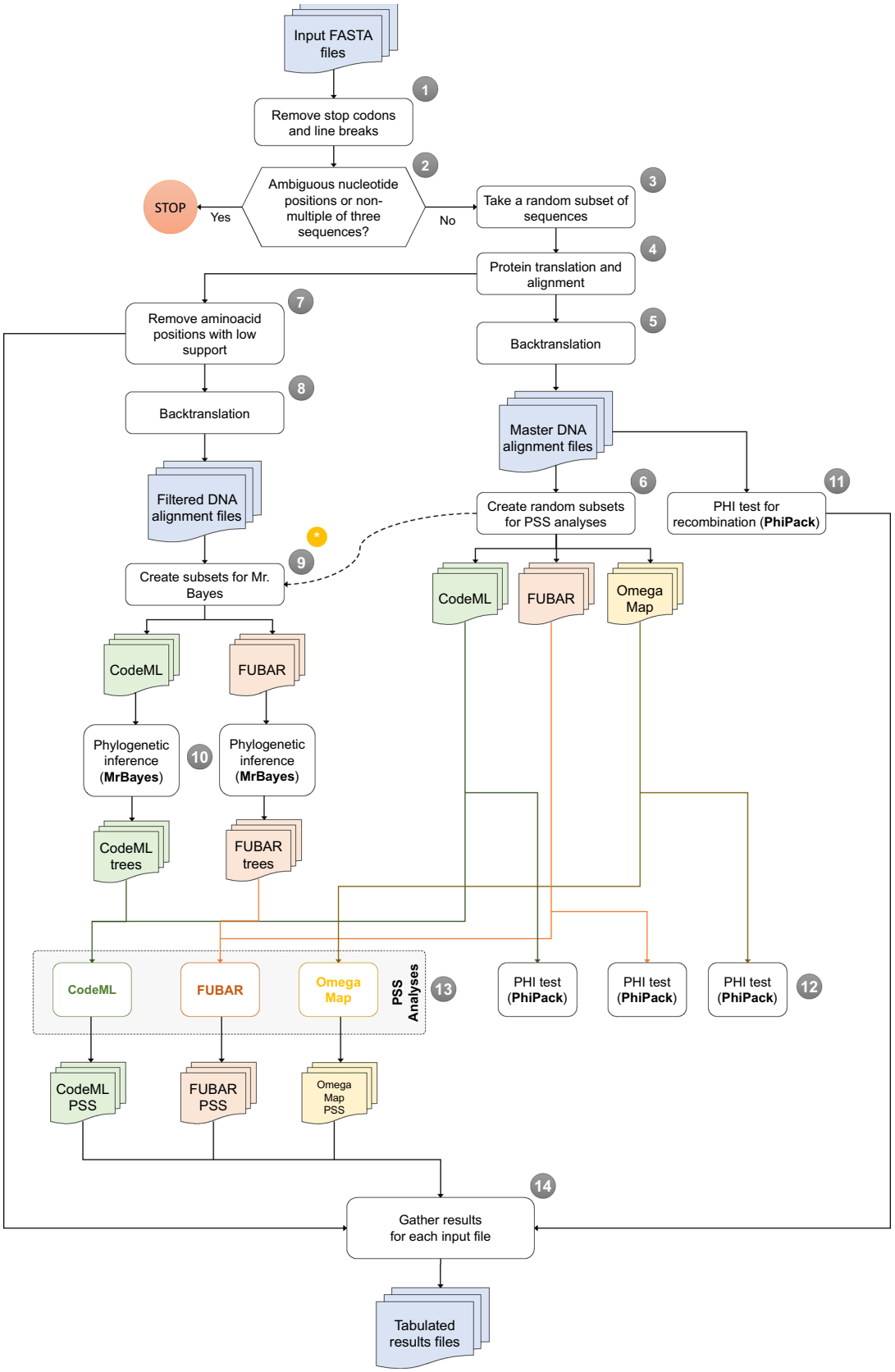



**Fig. 2** Steps and files involved in the IPSSA pipeline

FASTA file (step 3 in Fig. 2). The sequences in the master FASTA file are then aligned using the T-coffee suite [11]. Five alignment algorithms can be used: clustalw, muscle, kalign, t_coffee, and amap. Sequences are aligned at the amino acid level using the chosen alignment algorithm, and the corresponding nucleotide alignment obtained (steps 4 and 5 in Fig. 2). These are the master DNA alignment files (right branch on Fig. 2). The user can indicate how many times each method should be run, using a given number of sequences (also specified by the user for each method) sampled from the master alignment (step 6 in Fig. 2).

FUBAR and codeML require as input a phylogenetic tree describing the relationship between the sequences to be analysed. In IPSSA, the phylogenetic tree is obtained using MrBayes [12], using only codons that are aligned with a confidence score above a user-specified threshold (steps 7, 8, 9, and 10, left branch on Fig. 2). The model of sequence evolution that is used is the GTR (allowing for among-site rate variation and a proportion of invariable sites). Third codon positions are allowed to have a gamma distribution shape parameter that is different from that for first and second codon positions. Two independent runs of the number of generations specified by the user, with four chains each (one cold and three heated chains) are used. Trees are sampled every 100th generation. The first number of samples specified by the user are discarded (burn-in).

Since recombination can lead to the identification of false PSS, IPSSA also automatically runs the PhiPack software[8] that looks for evidence of recombination in the datasets. This is applied to the master DNA alignment files (step 11 in Fig. 2) as well as to the random subsets created for running each PSS detection method (step 12 in Fig. 2).

After running all PSS analyses (step 13 in Fig. 2), IPSSA creates a summary table (step 14 in Fig. 2) for each input file showing the sites identified by each method and runs, the location of sites showing alignment gaps, the location of sites with low support values in the alignment and whether there is evidence for recombination or not. All intermediate files generated by the underlying software used along with the output logs are saved and can be inspected by the user.

The source code of IPSSA is publicly available at GitHub[9] and the Docker image at Docker Hub.[10] As in the cases of FastScreen and GenomeFastScreen, IPSSA uses Docker images available at the pegi3s Docker Images Project[11] for running every third-party software required (codeML, omegaMap, FUBAR, T-Coffee, ALTER [13], seqkit [14], and so on).

[8] https://www.maths.otago.ac.nz/~dbryant/software/PhiPack.tar.

[9] https://github.com/pegi3s/ipssa.

[10] https://hub.docker.com/r/pegi3s/ipssa.

[11] https://pegi3s.github.io/dockerfiles/.

### 2.4 Data Source and Pre-processing

For *M. leprae*, we have used the set of 531 genes previously identified by us as likely showing PSS [5]. For *M. haemophilum*, the gene annotations of the five available genomes (GCF_000340435, GCF_001021405, GCF_001021415, GCF_001021435, and GCF_001021485) were downloaded from the NCBI Assembly RefSeq database on October 2020. GCF_000340435 was used as the *M. haemophilum* reference. Moreover, we downloaded from the NCBI Assembly RefSeq database the gene annotation of *M. tuberculosis* genome GCF_000195955 to be used as a global reference. Files downloaded from RefSeq were pre-processed using SEDA [15], to shorten header names (only accession numbers and gene names are kept).

For testing purposes regarding IPSSA running times (see Sect. 3.1.3), the FASTA file available at http://bpositive.i3s.up.pt/transcription?id=122416 was used.

### 2.5 Analyses

The IPSSA pipeline was used to analyse the 531 *M. leprae* genes that were previously identified as likely having PSS [5]. Moreover, here we identify the genes that likely show PSS in *M. haemophilum,* using the Auto-PSS-Genome pipeline. In both cases, muscle [16] was used as the alignment algorithm of choice, 1,000,000 generations and a burn-in of 25% was used for MrBayes, and all available methods for detecting PSS were used. Since for both *M. leprae* and *M. haemophilum,* a relatively small number of annotated genomes is available (less than 10), we have run each method once using all available sequences.

## 3 Results

### 3.1 Auto-PSS-Genome Running Times

First time users of the Auto-PSS-Genome pipeline (that involves running GenomeFastScreen, CheckCDS, and IPSSA) may be tempted to specify parameter values that imply very long running times. Therefore, in the following sections, we present the running times for the most time-consuming steps of the Auto-PSS-Genome pipeline.

#### 3.1.1 GenomeFastScreen

The GenomeFastScreen pipeline [5] takes a moderate-to-high amount of time to run. For instance, when running the *M. haemophylum* project (that involved the analysis of five *M. haemophylum* genomes plus one genome to be used as a global reference; see below), this task took about four and a half hours, when launching the pipeline with a maximum

**Table 1** Running times of the main steps involved in the IPSSA pipeline using a different number of sequences

| # Sequences | Alignment method's execution times (s) | | | | | MrBayes (min) | FUBAR (s) | codeml (h) | omegaMap (min) |
|---|---|---|---|---|---|---|---|---|---|
| | Clustalw | Muscle | Kalign | t_coffee | amap | | | | |
| 10 | 3 | 2 | 2 | 20 | 5 | 8.35 | 17 | 0.15 | 0.12 |
| 20 | 5 | 3 | 3 | 86 | 18 | 16.17 | 18 | 1.19 | 9.92 |
| 30 | 7 | 3 | 3 | 166 | 42 | 36.13 | 25 | **4.24** | 21.12 |
| 40 | 12 | 3 | 3 | 305 | 75 | 73.4 | 32 | 13.26 | 33.72 |
| 50 | 22 | 4 | 3 | 493 | 138 | 97.03 | 45 | 23.35 | 45 |
| 60 | 22 | 5 | 4 | 677 | 209 | 137.43 | 50 | n.a | 54.32 |
| 70 | 31 | 6 | 4 | 995 | 350 | 155.53 | 65 | n.a | 77.23 |
| 80 | 37 | 7 | 4 | 1281 | 413 | 182.7 | 67 | n.a | 102.67 |
| 90 | 48 | **8** | 5 | 1559 | 546 | **187.65** | **68** | n.a | **132.22** |

In bold-underline are the default values for IPSSA

50 parallel tasks in a computer with 96 CPUs (2.00 GHz) and 996 GB of RAM.

#### 3.1.2 CheckCDS

Although CheckCDS uses a greedy algorithm, when a small number of genomes is being analysed, only a small number of files must be processed by CheckCDS, and thus this step does not take long. For instance, when running the *M. haemophylum* project (see below), this task took 14 s only to be performed, when launching the pipeline with a maximum 50 parallel tasks in a computer with 96 CPUs (2.00 GHz) and 996 GB of RAM.

#### 3.1.3 IPSSA

Since the IPSSA pipeline is also useful for small-scale projects by itself, and not only in the context of the Auto-PSS-Genome pipeline, the following tests were run with a maximum one task on an average laptop with 7.7 GB of RAM and four 3.00 GHz CPUs. It should be noted that these values were obtained when not running any other process in parallel.

As can be seen in Table 1, even for a moderate number of sequences (e.g. 30), codeML takes almost four hours and a half to run. codeML needs as input the tree that is produced by MrBayes, that for 30 sequences takes about 36 min to be obtained. Therefore, running the codeML option alone can take more than 5 h even for a moderate number of sequences such as 30.

Although when using 90 sequences, FUBAR takes less than 70 s to run, results are produced after 3.1 h only, since this method also requires the tree produced by MrBayes. The only method that does not require a tree as input is omegaMap, but still takes about 2.2 h to finish the analysis of 90 sequences. Compared with the other processes, the time spent performing sequence alignments is negligible, unless T-coffee is used.

Based on these results, we have chosen as defaults for IPSSA the values shown in bold and underlined. This means that these values are used when the pipeline user does not provide values for them, either in the command line or in a configuration parameters file.

### 3.2 *Mycobacterium leprae* Results

When the 531 *M. leprae* genes that were previously identified as likely having PSS [5] are analysed using IPSSA, the following 26 genes are identified as being positively selected by codeML: *dnaA*, *dnaE*, *ftsX*, *gpsA*, *leuc*, *ML0051*, *ML0208*, *ML0240*, *ML0314*, *ML0606*, *ML0803*, *ML0825*, *ML1119*, *ML1182*, *ML1243*, *ML1286*, *ML1740*, *ML1750*, *ML2053*, *ML2570*, *ML2597*, *ML2630*, *ML2664*, *murE*, *recG*, and *tesB*. This number is similar to that obtained when using ADOPS [7] and the same set of input genes [5]. Only one gene (*ML1182*) out of the 26 genes has not been identified in the previous analyses. The differences reflect the stochastic nature of codeML. Since only 4.9% (26/531) of all gene sets identified by GenomeFastScreen as possibly showing PSS are identified, after detailed analyses, as having PSS, it seems unlikely that genes harbouring PSS are being excluded by GenomeFastScreen. Nevertheless, if surprised by the exclusion of a given gene from the dataset of genes worth to be analysed in detail, the user can run the standalone IPSSA pipeline here reported to analyse that gene in detail. On the other hand, if the user believes that PSS are identified at a given gene when only using the alignment algorithm that was specified, the IPSSA pipeline can also be used to quickly check the impact of using other alignment algorithms. The details of the IPSSA run are available as a 0.5 GB zip tar file (M_leprae.tar.xz file made available at https://doi.org/10.5281/zenodo.4279234).

Of the 26 genes identified by codeML as having PSS, 21 (*dnaA*, *ftsX*, *gpsA*, *leuc*, *ML0051*, *ML0208*, *ML0240*, *ML0606*, *ML0803*, *ML0825, ML1119*, *ML1243*, *ML1286*, *ML1740*, *ML2053*, *ML2570*, *ML2597*, *ML2664*, *murE*, *recG*, and *tesB*) are also identified by FUBAR as being positively selected, although for three of these (*ML0208*, *ML0825*, and *tesB*), codeML identified more PSS than FUBAR. For the remaining 23 cases, FUBAR and codeML identified the same amino acid sites as being positively selected. In only one case, FUBAR identified a gene as being positively selected that codeML did not (*glcB*), that was, however, previously identified as having PSS when using ADOPS [5], showing again the stochastic nature of codeML.

When running omegaMap, only 24 out of 531 genes did not show PSS. A possible explanation could be that omegaMap misbehaves, because the assumption that the individuals analysed come from the same population is being violated. Nevertheless, it is also possible that the observed low variation and recombination levels, coupled with the blocks approach used by omegaMap (oBlock and rBlock is set to 30 as recommended in omegaMap's manual) causes the observed problem. Indeed, for 529 out of 531 genes, inferences regarding recombination could not be performed when using PhiPack with a window size of 80 bp, because there are too few informative sites. For the two genes that could be analysed, there is no evidence for recombination.

When the biological functions of the genes that show PSS are taken into account, there are eight genes (*ML0051*, *ML0240*, *ML0314*, *ML0803*, *ML1243*, *ML2597*, *recG*, and *tesB*, that are the orthologues of *M. tuberculosis PPE68*, *rpfB*, *lipU*, *Rv3220c*, *lipQ*, *Rv0177*, *recG*, and *tesB1* genes) for which the phenotypic traits being positively selected are likely the modulation of the host inflammatory response (*Rv0177*; [17]), the establishment and maintenance of infection (*PPE68* [18];), resuscitation from dormancy (*rpfB*; [19]), protection against mitomycin C, methyl methane sulfonate and UV induced cell death (*recG*; [20]), as well as survival, persistence, and virulence (*lipQ, lipU, Rv3220c,* and *tesB1*; [21]), discussed in detail in [5]. Of these, only *ML0314* has been identified by codeML only (see above), suggesting that genes identified by both FUBAR and codeML as having PSS may indeed harbour amino acid variation that may have an impact on important *M. leprae* phenotypic traits.

### 3.3 *Mycobacterium haemophilum* Results

Out of the 3884 orthologous gene sets identified by the GenomeFastScreen step of the Auto-PSS-Genome pipeline, 1179 were identified as deserving further detailed analyses. For these, the details of the IPSSA run are available as a 1.4 GB zip tar file (M_haemophylum.tar.xz file made available at https://doi.org/10.5281/zenodo.4279234). Out of the 1179 gene sets analysed, FUBAR identified 184 *M. haemophilum* genes as being positively selected. Nevertheless, codeML identified 47 genes only as being positively selected (Supplementary Table 1). There are 24 genes in common between the two datasets (*asnB*, *cobN*, *mfd*, *secA*, *nrdI*, *RS03750*, *RS06295*, *RS07170*, *RS08670*, *RS08955*, *RS09760*, *RS09875*, *RS11065*, *RS13820*, *RS14355*, *RS14775*, *RS16965*, *RS17140*, *RS18085*, *RS18250*, *RS19140*, *RS19480*, *RS19720*, and *RS20035*; gene names are for the *M. haemophilum* strain ATCC 29548 (ASM34043v3), the one used as a local reference). For these, the evidence for PSS is strongest. In the set of 24 genes where PSS have been identified by both methods, FUBAR identified 46 PSS of which 44 were also identified by codeML. Nevertheless, codeML identified many more PSS (370) than FUBAR (46). This result stands, even if the genes where codeML identified more than 10 PSS are viewed as suspicious and, therefore, discarded, in which case the total would be 88.

As for *M. leprae,* when using omegaMap, 1093 out of 1179 gene sets (92.7%) showed evidence for PSS. Moreover, for most gene sets, many PSS were identified. As pointed out above, the reasons for the omegaMap misbehaviour may be the violation of the assumption that the individuals come from the same population, or the observed low variation and recombination levels coupled with the blocks approach used by omegaMap (oBlock and rBlock is set to 30 as recommended in omegaMap's manual). Indeed, for 1171 gene sets, inferences regarding recombination could not be performed using PhiPack with a window size of 80 bp, because there are too few informative sites. For the eight genes that could be analysed, there is no evidence for recombination.

In *M. tuberculosis*, the orthologous genes of the 24 genes *M. haemophilum* dataset, as determined by GenomeFastScreen are, respectively: *asnB*, *cobN*, *mfd*, *secA1*, *nrdI*, –, *Rv1354c*, *dxr*, –, *Rv1433*, *Rv1638A*, *pks7*, –, –, –, –, *Rv0791c*, –, –, *Rv3675*, *embB*, *esxA*, –, and –, where "–" means that no orthologous gene was identified. When the biological function of these genes in *M. tuberculosis* is taken into account, for five genes (*asnB*, *dxr*, *pks7, embB,* and *esxA*) the phenotypic trait being positively selected can be inferred as likely being resistance to drugs and virulence.

Mycobacteria are naturally resistant to multiple drugs and *asnB* plays a role in the setting of this natural resistance [22]. Therefore, although the codeML and FUBAR analyses do not agree on the identified PSS (codeML identifies alignment positions 35 and 49 as PSS, while FUBAR identifies position 66 as a PSS), it is conceivable that changes at these amino acid positions do play a role in the resistance to drugs used in the treatment of *M. haemophilum* infections.

In *Mycobacteria, dxr* is responsible for the intrinsic resistance to fosmidomycin [23]. Therefore, as in the above case, it may play a role in the resistance to drugs used in the treatment of *M. haemophilum* infections. codeML identified

five alignment positions as being PSS (11, 50, 75, 233 and 264), while FUBAR identified one (75).

The *pks7* gene seems to be involved in the synthesis of phthiocerol dimycocerosates [24] and is a well-known virulence factor in *M. tuberculosis* [25]. In the *M. tuberculosis* Beijing family, that is of interest since it is increasingly associated with drug resistance throughout the world, there is a lineage specific amino acid replacement at this gene [25]. Here, for *M. haemophilum*, codeML identified eight alignment positions as being PSS (1160, 1442, 1443, 1445, 1493, 1618, 1772, and 2063) of which FUBAR identified five (1160, 1443, 1445, 1493, and 1772). These positions may thus be associated with different virulence levels.

In clinical *M. tuberculosis* isolates, missense mutations in *embB* are associated with Ethambutol resistance [26], and this drug is used in the treatment of *M. haemophilum* infections (see for instance [8]). Therefore, the four PSS (positions 14, 16, 140, and 265 in the alignment) identified by codeML (three of which were also identified by FUBAR, namely positions 14, 16, and 265 in the alignment) may be clinically relevant.

The EsxA protein is also a major virulence factor of *M. tuberculosis* [27]. Both codeML and FUBAR identified alignment positions 28 and 36 as PSS. Therefore, these positions may also be associated with different virulence levels.

It should be noted that there is little to no overlap between the genes inferred to be under positive selection in *M. leprae* and *M. haemophilum* despite being two closely related species. Indeed, in the analyses here presented there is no overlap, although there is a two gene (*mfd* and *Rv1354c*) overlap between the results here obtained for *M. haemophilum* and a previous analysis performed for *M. leprae* [5]. The relevance of such genes in the context of resistance to drugs is, however, unknown. Therefore, it seems that *M. leprae* and *M. haemophilum* are under very different selective pressures, which may be expected since they cause very different diseases.

## 4 Conclusion

Auto-PSS-genome allows the identification of genes showing PSS, using multiple methods, quickly and almost without user intervention, starting from FASTA files, one per genome, containing all annotated coding sequences. We show the usefulness of such a pipeline using *M. leprae* and *M. haemophilum*. A Docker image is made available for the Auto-PSS-Genome pipeline together with detailed instructions on how to use it, and thus even researchers without a background in informatics should be able to easily run it.

**Supplementary Information** The online version contains supplementary material available at https://doi.org/10.1007/s12539-021-00439-2.

**Acknowledgements** This work was funded by National Funds through FCT—Fundação para a Ciência e a Tecnologia, I.P., under the project UIDB/04293/2020. The SING group thanks the CITI (Centro de Investigación, Transferencia e Innovación) from the University of Vigo for hosting its IT infrastructure. This work was partially supported by the Consellería de Educación, Universidades e Formación Profesional (Xunta de Galicia) under the scope of the strategic funding ED431C2018/55-GRC Competitive Reference Group.

## Declarations

**Conflict of interests** On behalf of all authors, the corresponding author states that there is no conflict of interest.

## References

1. Yang Z (1997) PAML: a program package for phylogenetic analysis by maximum likelihood. Bioinformatics 13:555–556. https://doi.org/10.1093/bioinformatics/13.5.555
2. Murrell B, Moola S, Mabona A, Weighill T, Sheward D, Kosakovsky Pond SL, Scheffler K (2013) FUBAR: a fast, Unconstrained Bayesian AppRoximation for inferring selection. Mol Biol Evol 30:1196–1205. https://doi.org/10.1093/molbev/mst030
3. Wilson DJ, McVean G (2006) Estimating diversifying selection and functional constraint in the presence of recombination. Genetics 172:1411–1425. https://doi.org/10.1534/genetics.105.044917
4. López-Fernández H, Duque P, Vázquez N, Fdez-Riverola F, Reboiro-Jato M, Vieira CP, Vieira J (2020) Inferring positive selection in large viral datasets. In: Fdez-Riverola F, Rocha M, Mohamad MS, Zaki N, Castellanos-Garzón JA (eds) Practical applications of computational biology and bioinformatics, 13th international conference. Springer, Cham, pp 61–69. https://doi.org/10.1007/978-3-030-23873-5_8
5. López-Fernández H, Vieira CP, Fdez-Riverola F, Reboiro-Jato M, Vieira J (2021) Inferences on *Mycobacterium Leprae* host immune response escape and antibiotic resistance using genomic data and GenomeFastScreen. In: Panuccio G, Rocha M, Fdez-Riverola F, Mohamad MS, Casado-Vara R (eds) Practical applications of computational biology and bioinformatics, 14th international conference (PACBB 2020). Springer, Cham, pp 42–50. https://doi.org/10.1007/978-3-030-54568-0_5
6. Osório NS, Rodrigues F, Gagneux S, Pedrosa J, Pinto-Carbó M, Castro AG, Young D, Comas I, Saraiva M (2013) Evidence for diversifying selection in a set of mycobacterium tuberculosis genes in response to antibiotic- and nonantibiotic-related pressure. Mol Biol Evol 30:1326–1336. https://doi.org/10.1093/molbev/mst038
7. Reboiro-Jato D, Reboiro-Jato M, Fdez-Riverola F, Vieira CP, Fonseca NA, Vieira J (2012) ADOPS—automatic detection of positively selected sites. J Integr Bioinform 9:200. https://doi.org/10.2390/biecoll-jib-2012-200
8. Lindeboom JA, van Coppenraet LESB, van Soolingen D, Prins JM, Kuijper EJ (2011) Clinical manifestations, diagnosis, and treatment of *Mycobacterium haemophilum* infections. Clin Microbiol Revi 24:701–717. https://doi.org/10.1128/CMR.00020-11
9. Pin D, Guérin-Faublée V, Garreau V, Breysse F, Dumitrescu O, Flandrois J-P, Lina G (2012) *Mycobacterium* species related to *M. leprae* and *M. lepromatosis* from cows with bovine nodular thelitis. Emerg Infect Dis 20:2111–2114. https://doi.org/10.3201/eid2012.140184
10. Sievers F, Higgins DG (2018) Clustal omega for making accurate alignments of many protein sequences: clustal omega for many

protein sequences. Protein Sci 27:135–145. https://doi.org/10.1002/pro.3290
11. Denoeud F, Carretero-Paulet L, Dereeper A, Droc G, Guyot R, Pietrella M, Zheng C, Alberti A, Anthony F, Aprea G, Aury J-M, Bento P, Bernard M, Bocs S, Campa C, Cenci A, Combes M-C, Crouzillat D, Silva CD, Daddiego L, Bellis FD, Dussert S, Garsmeur O, Gayraud T, Guignon V, Jahn K, Jamilloux V, Joët T, Labadie K, Lan T, Leclercq J, Lepelley M, Leroy T, Li L-T, Librado P, Lopez L, Muñoz A, Noel B, Pallavicini A, Perrotta G, Poncet V, Pot D, Priyono A, Rigoreau M, Rouard M, Rozas J, Tranchant-Dubreuil C, VanBuren R, Zhang Q, Andrade AC, Argout X, Bertrand B, de Kochko A, Graziosi G, Henry RJ, Jayarama S, Ming R, Nagai C, Rounsley S, Sankoff D, Giuliano G, Albert VA, Wincker P, Lashermes P (2014) The coffee genome provides insight into the convergent evolution of caffeine biosynthesis. Science 345:1181–1184. https://doi.org/10.1126/science.1255274
12. Ronquist F, Teslenko M, van der Mark P, Ayres DL, Darling A, Höhna S, Larget B, Liu L, Suchard MA, Huelsenbeck JP (2012) MrBayes 3.2: efficient Bayesian phylogenetic inference and model choice across a large model space. Syst Biol 61:539–542. https://doi.org/10.1093/sysbio/sys029
13. Glez-Peña D, Gómez-Blanco D, Reboiro-Jato M, Fdez-Riverola F, Posada D (2010) ALTER: program-oriented conversion of DNA and protein alignments. Nucleic Acids Res 38:W14-18. https://doi.org/10.1093/nar/gkq321
14. Shen W, Le S, Li Y, Hu F (2016) SeqKit: a cross-platform and ultrafast toolkit for FASTA/Q file manipulation. PLoS ONE 11:e0163962. https://doi.org/10.1371/journal.pone.0163962
15. López-Fernández H, Duque P, Henriques S, Vázquez N, Fdez-Riverola F, Vieira CP, Reboiro-Jato M, Vieira J (2019) Bioinformatics protocols for quickly obtaining large-scale data sets for phylogenetic inferences. Interdiscip Sci Comput Life Sci 11:1–9. https://doi.org/10.1007/s12539-018-0312-5
16. Edgar RC (2004) MUSCLE: multiple sequence alignment with high accuracy and high throughput. Nucleic Acids Res 32:1792–1797. https://doi.org/10.1093/nar/gkh340
17. Shimono N, Morici L, Casali N, Cantrell S, Sidders B, Ehrt S, Riley LW (2003) Hypervirulent mutant of *Mycobacterium tuberculosis* resulting from disruption of the mce1 operon. Proc Natl Acad Sci 100:15918–15923. https://doi.org/10.1073/pnas.2433882100
18. Demangel C, Brodin P, Cockle PJ, Brosch R, Majlessi L, Leclerc C, Cole ST (2004) Cell envelope protein PPE68 contributes to *Mycobacterium tuberculosis* RD1 Immunogenicity Independently of a 10-kilodalton culture filtrate protein and ESAT-6. Infect Immun 72:2170–2176. https://doi.org/10.1128/IAI.72.4.2170-2176.2004
19. Squeglia F, Romano M, Ruggiero A, Vitagliano L, De Simone A, Berisio R (2013) Carbohydrate recognition by RpfB from *Mycobacterium tuberculosis* unveiled by crystallographic and molecular dynamics analyses. Biophys J 104:2530–2539. https://doi.org/10.1016/j.bpj.2013.04.040
20. Thakur RS, Basavaraju S, Somyajit K, Jain A, Subramanya S, Muniyappa K, Nagaraju G (2013) Evidence for the role of *Mycobacterium tuberculosis* RecG helicase in DNA repair and recombination. FEBS J 280:1841–1860. https://doi.org/10.1111/febs.12208
21. Li C, Li Q, Zhang Y, Gong Z, Ren S, Li P, Xie J (2017) Characterization and function of *Mycobacterium tuberculosis* H37Rv Lipase Rv1076 (LipU). Microbiol Res 196:7–16. https://doi.org/10.1016/j.micres.2016.12.005
22. Ren H, Liu J (2006) AsnB is involved in natural resistance of *Mycobacterium smegmatis* to multiple drugs. AAC 50:250–255. https://doi.org/10.1128/AAC.50.1.250-255.2006
23. Brown AC, Parish T (2008) Dxr is essential in *Mycobacterium tuberculosis* and fosmidomycin resistance is due to a lack of uptake. BMC Microbiol 8:78. https://doi.org/10.1186/1471-2180-8-78
24. Virulence attenuation of two Mas-like polyketide synthase mutants of *Mycobacterium tuberculosis* | Microbiology Society. https://www.microbiologyresearch.org/content/journal/micro/10.1099/mic.0.26278-0. Accessed 13 Nov 2020
25. Koster K, Largen A, Foster JT, Drees KP, Qian L, Desmond EP, Wan X, Hou S, Douglas JT (2018) Whole genome SNP analysis suggests unique virulence factor differences of the Beijing and Manila families of *Mycobacterium tuberculosis* found in Hawaii. PLoS ONE 13:e0201146. https://doi.org/10.1371/journal.pone.0201146
26. Starks AM, Gumusboga A, Plikaytis BB, Shinnick TM, Posey JE (2009) Mutations at embB Codon 306 are an important molecular indicator of ethambutol resistance in *Mycobacterium tuberculosis*. AAC 53:1061–1066. https://doi.org/10.1128/AAC.01357-08
27. Chen JM, Zhang M, Rybniker J, Boy-Röttger S, Dhar N, Pojer F, Cole ST (2013) *Mycobacterium tuberculosis* EspB binds phospholipids and mediates EsxA-independent virulence. Mol Microbiol 89:1154–1166. https://doi.org/10.1111/mmi.12336