SCIENTIFIC REPORTS

OPEN

Predicting synonymous codon usage and optimizing the heterologous gene for expression in *E. coli*

Received: 19 January 2017
Accepted: 11 August 2017
Published online: 30 August 2017

Jian Tian[1], Yaru Yan[2,1], Qingxia Yue[3,1], Xiaoqing Liu[1], Xiaoyu Chu[1], Ningfeng Wu[1] & Yunliu Fan[1]

Of the 20 common amino acids, 18 are encoded by multiple synonymous codons. These synonymous codons are not redundant; in fact, all of codons contribute substantially to protein expression, structure and function. In this study, the codon usage pattern of genes in the *E. coli* was learned from the sequenced genomes of *E. coli*. A machine learning based method, Presyncodon was proposed to predict synonymous codon selection in *E. coli* based on the learned codon usage patterns of the residue in the context of the specific fragment. The predicting results indicate that Presycoden could be used to predict synonymous codon selection of the gene in the *E. coli* with the high accuracy. Two reporter genes (*egfp* and *mApple*) were designed with a combination of low- and high-frequency-usage codons by the method. The fluorescence intensity of eGFP and mApple expressed by the (*egfp* and *mApple*) designed by this method was about 2.3- or 1.7- folds greater than that from the genes with only high-frequency-usage codons in *E. coli*. Therefore, both low- and high-frequency-usage codons make positive contributions to the functional expression of the heterologous proteins. This method could be used to design synthetic genes for heterologous gene expression in biotechnology.

In naturally occurring genes, 61 codons code for the 20 common amino acids. The role of synonymous codons is unclear, as they do not alter the encoded amino acid sequence[1]. Therefore, it was initially thought that they would not affect cellular function, organismal fitness or evolution[2,3]. However, several studies have found that synonymous codon selection in a gene could affect the expression[4–6], structure and function of the encoded protein[7–9]. Therefore, it is useful to know the rules governing synonymous codon selection of the target gene, as such knowledge could enable us to design the heterogenous gene with the most efficient expression in the expression host.

The synonymous sequences contain varying ratios of low-frequency-usage (i.e., more slowly translated) to high-frequency-usage codons, which could control the translation speed of a protein[10,11]. If a structural element within a protein is not translated with the appropriate speed, it can affect the folding the synthesized protein fragment and the assembly the structural elements of the protein[7]. Several studies have examined the overall translational rate of various protein structural elements[12–15]. For example, high-frequency-usage codons are mainly associated with α-helices. However, lower-frequency-usage codons are more likely to be associated with β-strands, random coils, structural domain boundaries and trans-membrane helices[7,12,14,16]. The translation speed decreases on transition from coil to helix or strand[13]. As a result, the synonymous codons could affect the kinetics of translation and regulate the timing of protein synthesis at the local or global scales[10,17]. In addition to the synonymous codon usage, many studies found that the codon pair usage, also known as codon context could affect the protein expression in the host[18–20]. Therefore, to express a target gene efficiently in a heterologous expression system, the gene should be designed based on the codon usage pattern of the host strain of the expression system and the codon selection constraints of the target gene[21–24].

Many methods, including COOL[25], Gene Designer[26], Gene composer[27], JCat[28], COStar[29] and OPTIMIZER[30], have been proposed to design heterologous genes which are expected to be efficiently expressed in the host organism. Based on our knowledge, these methods are prone to select the high-frequency-usage codons of the host

[1]Biotechnology Research Institute, Chinese Academy of Agricultural Sciences, Beijing, 100081, China. [2]College of Food Science and Technology, Agricultural University of Hebei, Baoding, HeBei Provice, 071001, China. [3]Institute of Microbial Biotechnology, Jinan University, Guangzhou, Guangdong Province, 510632, China. Correspondence and requests for materials should be addressed to N.W. (email: wuningfeng@caas.cn)

for the heterologous gene and neglect the contribution of the low-frequency-usage codons (rare codons) to the expression of the target gene. These approaches have been successfully used for the heterologous production of some proteins, especially the proteins also encoded by the "preferred" codons in the native host[31,32]. However, in some cases, the high levels of protein expressed with high speed translation from the N- to C-terminal have led to the formation of insoluble products or degradation by protease enzymes, due to the incorrect folding[33,34].

The low expression or formation of insoluble aggregates may be attributable to differences in synonymous codon bias between the expression and natural hosts[35]. Another recent approach to encode a target gene sequence in the heterologous host is to "match" the codon usage bias inherent in the native host and is referred to as "codon harmonization"[14,35,36]. This codon harmonization approach was successfully applied to express several proteins in *E. coli*[35–37]. However, if we did not know the codon usage bias of the native host, such as the gene cloned from the metagenome, this method could not be used. In addition, it is difficult to perfectly "match" the codon usage bias of the native host to the expression host. Therefore, a method should be developed to design the heterogenous gene with the appropriate synonymous codons in the expression host.

In this study, data from bacterial genomes in GenBank were used to analyze the rules of the synonymous codon selection in *Escherichia coli*. The codon usage pattern of a residue within a specific fragment was learned from all *E. coli* genomes in the GenBank, and this information was stored in index files. Based on those index files, a machine-learning method named Presyncodon was developed to predict synonymous codon (low- or high-frequency-usage codon) selection in a gene. The two reporter genes, encoding enhanced green fluorescent protein (*egfp*) and red fluorescent protein (*mApple*), were designed with the method. Expression of the designed genes yielded a higher fluorescence intensity compared with the genes in which low-frequency-usage codons had been replaced by high-frequency-usage codons. This result revealed that both low- and high-frequency-usage codons make positive contributions to the solubility of expressed recombinant proteins. In addition, this study will help us to understand codon selection rules and design genes that are more amenable to heterologous expression.

Results

Codon Usage Patterns of Different Bacterial Species. The sequences of 346 genomes of bacterial subspecies were collected from the NCBI database (Table S1), including five subspecies from 69 bacterial species and one subspecies from one species (*Bacteroides fragilis* NCTC 9343) selected as the out-group[38]. This dataset allowed us to easily evaluate the clustering results, as the five subspecies should be clustered into one group. The selected bacteria represented 38 families, 47 genera and 70 species. An evolutionary tree based on the 16 S rDNA of these subspecies was constructed. The 16 S rDNA tree in the Newick format is shown in Fig. S1. As shown in Figs S2A and S3, the five subspecies from each of the species were clustered into one group except for those from two closely related genera, *Bacillus* (*B. anthracis*, *B. cereus* and *B. thuringiensis*) and *Mycobacterium* (*M. canettii*, *M. bovis* and *M. tuberculosis*), which cross-clustered into one large group. In addition, species within the same genus and family clustered together.

The codon usage pattern of each subspecies was calculated, normalized and clustered (Fig. 1), and an evolutionary tree based on the codon usage patterns was constructed (Figs S2B and S4). The tree of those selected bacterial genomes in the Newick format is shown in Fig. S5. Nearly all of the subspecies within each species clustered into one group based on the codon usage patterns. Most of the species showed a higher usage frequency for the codons ATG, GAT and GAA than for other codons. However, the usage frequency for the codons CTA, AGG and CGA in these bacteria was lower than for other codons. Except for the six codons ATG, GAT, GAA, CTA, AGG and CGA, there was considerable deviation in codon usage patterns between bacteria belonging to different taxa. Bacterial strains within the same species had similar codon usage patterns. However, as shown in Figs S4 and S5, species within the same genus (*Bacillus*, *Lactobacillus*, *Mycoplasma* or *Corynebacterium*) showed different codon usage patterns. For example, as shown in Fig. S4, the codon usage pattern of *Bacillus amyloliquefaciens* was very similar to that of *B. subtilis* but differed greatly from those of *B. anthracis*, *B. cereus* and *B. thuringiensis*. Therefore, if a target gene is isolated from a different species or genus from the host used in a heterologous expression system, the gene may need to be optimized for the efficient expression in the host strain.

Codon Usage Pattern of the Middle Amino Acid in short peptides. All of the protein sequences encoded by the 65 genomes of *E. coli* (Table S1) were split into window sizes of one, three, five or seven amino acids. The same amino acid fragments from all of the genomes were merged, and the codon usage distribution of the middle amino acid in the fragment was calculated. The codon usage entropy of each amino acid was calculated, which represents the uncertainty of the codon selection of the amino acid. If the entropy of an amino acid is 1, the synonymous codon would be randomly selected. If the entropy is 0, however, only one specific codon can be selected for the amino acid. As shown in Fig. 2, the middle amino acid in the fragments with five or seven residues is likely to be coded by one specific codon, as the entropies for these fragments were significantly lower than those for fragments with one or three residues. As shown in Fig. 3, the codon of the middle residue in some fragments was determined by the peptide-dependent selection of the specific codon. For example, the rarest codon among the 61 codons in *E. coli* is AGG, which accounted for only 1.4‰ of the codons for arginine in the 65 *E. coli* genomes. There were 132 fragments containing the sequence GRRVA in the translated genes from the 65 *E. coli* genomes, and all of the 132 codons for the middle residue (arginine) were AGG. Therefore, the middle amino acid within a short peptide with length greater and equal than five amino acids usually used the specific codon by the peptide to code the residue. And the codon usage pattern of the short peptide will be used as the input vector of Presnycodon.

Codon Prediction with the Machine-Learning Method. The *E. coli* genome dataset (Table S1) contains 65 genomes, 64 of which were selected to create a codon selection index (CSI) file. To eliminate the over-fit effect

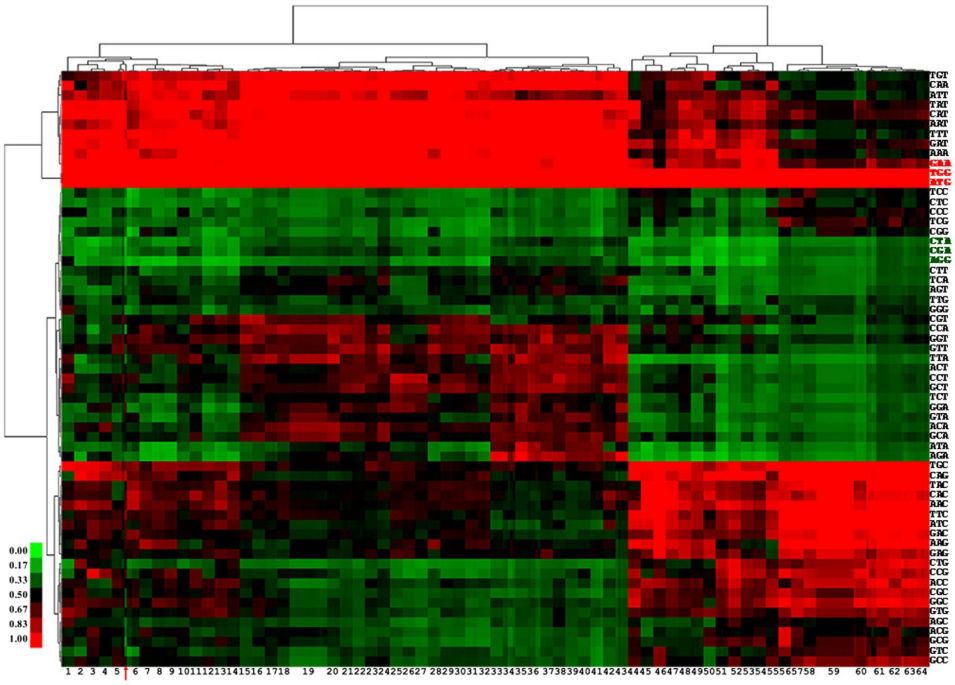

Figure 1. Clustering results of the codon usage pattern of different species. The row and column represent the codon usage pattern and the different bacterial subspecies. The species between species id 5 and 6 (red arrow) is Bacteroides fragilis NCTC 9343. The numbers from 1 to 64 refer to the bacterial genera. 1 Helicobacter pylori, 2 Acetobacter pasteurianus, 3 Bacillus amyloliquefaciens, 4 Bacillus subtilis, 5 Zymomonas mobilis, 6 Alteromonas macleodii, 7 Lactobacillus plantarum, 8 Lactobacillus casei, 9 Lactobacillus rhamnosus, 10 Coxiella burnetii, 11 Mannheimia haemolytica, 12 Shewanella baltica, 13 Vibrio cholerae, 14 Yersinia pestis, 15 Acinetobacter baumannii, 16 Haemophilus influenzae, 17 Enterococcus faecalis, 18 Listeria monocytogenes, 19 Bacillus anthracis, Bacillus cereus or Bacillus thuringiensis, 20 Staphylococcus aureus, 21 Lactococcus lactis, 22 Streptococcus agalactiae, 23 Legionella pneumophila, 24 Mycoplasma hyopneumoniae, 25 Chlamydia trachomatis, 26 Chlamydophila pneumoniae, 27 Chlamydophila psittaci, 28 Lactobacillus reuteri, 29 Streptococcus dysgalactiae, 30 Streptococcus pyogenes, 31 Streptococcus pneumoniae, 32 Streptococcus suis, 33 Borrelia burgdorferi, 34 Prochlorococcus marinus, 35 Clostridium botulinum, 36 Candidatus Kinetoplastibacterium, 37 Francisella tularensis, 38 Campylobacter jejuni, 39 Rickettsia prowazekii, 40 Rickettsia rickettsii, 41 Wolbachia endosymbiont, 42 Mycoplasma gallisepticum, 43 Mycoplasma hyorhinis, 44 Brucella melitensis, 45 Corynebacterium glutamicum, 46 Propionibacterium acnes, 47 Corynebacterium diphtheria, 48 Corynebacterium pseudotuberculosis, 49 Xylella fastidiosa, 50 Treponema pallidum, 51 Enterobacter cloacae, 52 Klebsiella pneumoniae, 53 *Escherichia coli*, 54 Salmonella enterica, 55 Neisseria meningitidis, 56 Burkholderia pseudomallei, 57 Bifidobacterium animalis, 58 Bifidobacterium longum, 59 Mycobacterium bovis, Mycobacterium canettii or Mycobacterium tuberculosis, 60 Rhodopseudomonas palustris, 61 Pseudomonas fluorescens or Pseudomonas aeruginosa, 62 Ralstonia solanacearum, 63 Pseudomonas putida, 64 Pseudomonas stutzeri.

of the model, one of which (*E. coli* K12_MG1655) was excluded in the construction of the CSI file was used to evaluate performance of the method and. All of the proteins encoded by the 64 genomes of *E. coli* were split into window sizes of three, five or seven amino acids. The same amino acid fragments from all of the genomes were merged into one CSI vector, which contained the codon usage distribution for the middle amino acid and the average codon usage for each amino acid in the fragment. Thus, 8000, 1,686,761 and 4,366,175 CSI vectors were generated for the three-, five- and seven-amino acid *E. coli* fragments, respectively.

All of the translated protein sequences from *E. coli* K12_MG1655 were split into window sizes of three, five or seven amino acids and searched against the corresponding CSI files for *E. coli*. The entirely matched index record was used to generate the input vector to predict the codon selection of the middle amino acid. As there are 18 amino acids coded by multiple codons in nature, 18 classifiers for each window size were constructed to predict the codon selection of the target amino acid. A ten-fold cross validation was carried out to evaluate the performance of each classifier. As shown in Fig. S7, if every codon was predicted as the high-frequency-usage codon, the median prediction accuracy for each amino acid was ~54.3%. However, if the window size was five or seven amino acids, the median prediction accuracy of the 18 classifiers increased to 80.53 and 97.54%, respectively, as shown in Fig. S7.

Thus, if a window size of five or seven amino acids was selected, the classifier could obtain high accuracy. However, it was based on only 1,686,761 and 4,366,175 *E. coli* CSI values for the five- and seven-amino acid windows, which is only 52.7 and 0.3% of all of the possible values for five ($20^5 = 3{,}200{,}000$) and seven

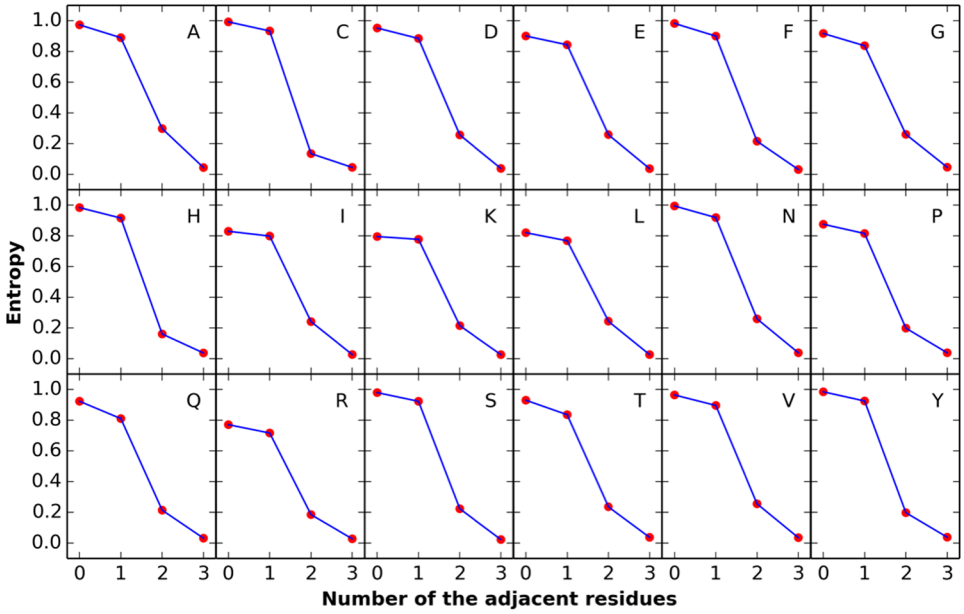

Figure 2. The entropy of the codon usage pattern of the middle amino acid with the different amino acid neighbors in *E. coli*. The x-axis represents the different number of the adjacent amino acids. The y-axis represents the average entropy of all codon usage pattern of the middle amino acid with corresponding adjacent amino acids. The data were calculated by the 65 genomes of *E. coli* (Table S1).

($20^7 = 1{,}280{,}000{,}000$) amino acid windows. Here, we didn't consider the longer window size than 7 amino acids, as the long window size contained more possible amino acid fragments. If the fragments being assessed were not in the index file, the target codon would not be predicted. To predict codon usage for the most fragments possible, the matched percent (p, $p = s/m$) for each fragment was assessed, which is the percent of matching between a calculated matched score (s) and expected maximal score (m) of the target fragment, as shown in Fig. S6. For a given cutoff (c), if the matched percent of multiple fragments from the CSI files was greater than the cutoff, the coding vector for the target codon is just the arithmetic average of all of the matched record vectors. The evaluation results for the different cutoffs are shown in Fig. 4 and Table S2. The classifier achieved high accuracy when the c was greater than 0.9 for a window size of five amino acids or greater than 0.8 for a window size of seven amino acids, which the AUCs (Area Under the receiver operating characteristic Curve) of most of those classifiers were great than 0.7 and 0.8, respectively.

The numbers of the fragments of five and seven amino acids in all 346 bacterial genomes were 2,758,946 and 66,114,871, respectively. If the cutoff (c) was set as 0.8, 99.3 and 63.8% of the five- and seven-amino acid fragments, respectively, could be predicted by the method. Therefore, based on this idea, most of the codons in a heterologous gene could be predicted by the method with the appropriate cutoff (c) and window size.

For the aim to predict the codon selection of a target gene, all predicting models with different window sizes (5 and 7 amino acids) and cutoff c (0.8, 0.85, 0.9, 0.95 and 1) were constructed. The models with window size of seven amino acids and big cutoff c have priority over the models with window size of five amino acids and small cutoff c, respectively. Based on this process, each amino acid was predicted the codon usage tendency by only one predicting model which the long window size and big cutoff c should be selected with priority. As a result, the gene sequence of a target protein was predicted from the amino acid sequence by the method except the first and last two codons of the gene. As the first 30 codons of a gene usually selected as the codons with the low-frequency-usage codons[5,39], the first two codons of the gene were selected as the low-frequency-usage codons. The last two codons usually select the high-frequency-usage codons, as they didn't affect the expression of the target genes.

Design of a Codon-Optimized Reporter Gene. The two reporter genes (*egfp-codon* and *mApple-codon*) were designed using the classifiers described above. In addition, another two control genes (*egfp-genscript* and *mApple-genscript*) were designed, which mainly used the high-frequency-usage codons of *E. coli* designed by the GenScript software. The sequences of those genes are shown in Figs S8 and S9. The four genes (*egfp-codon*, *mApple-codon, egfp-genscript* and *mApple-genscript*) were expressed in the same *E. coli* expression system. The fluorescence intensity of eGFP and mApple expressed from eGFP-codon and mApple-codon was about 2.3- or 1.7-folds greater than that from eGFP-genscript and mApple-genscript (Fig. 5). In addition, as shown in SDS-PAGE (Fig. S10), the amount of the expressed proteins eGFP and mApple from eGFP-codon and mApple-codon was also higher than that from eGFP-genscript and mApple-genscript. As shown in the codon usage table for the four genes in Table S3, there are several low-frequency-usage codons in the designed genes (*egfp-codon* and *mApple-codon*), such as CTA, CGA and AGA. The designed reporter genes (*egfp-codon* and *mApple-codon*) are a combination of low- and high-frequency-usage codons. Therefore, the low-frequency-usage codons also make

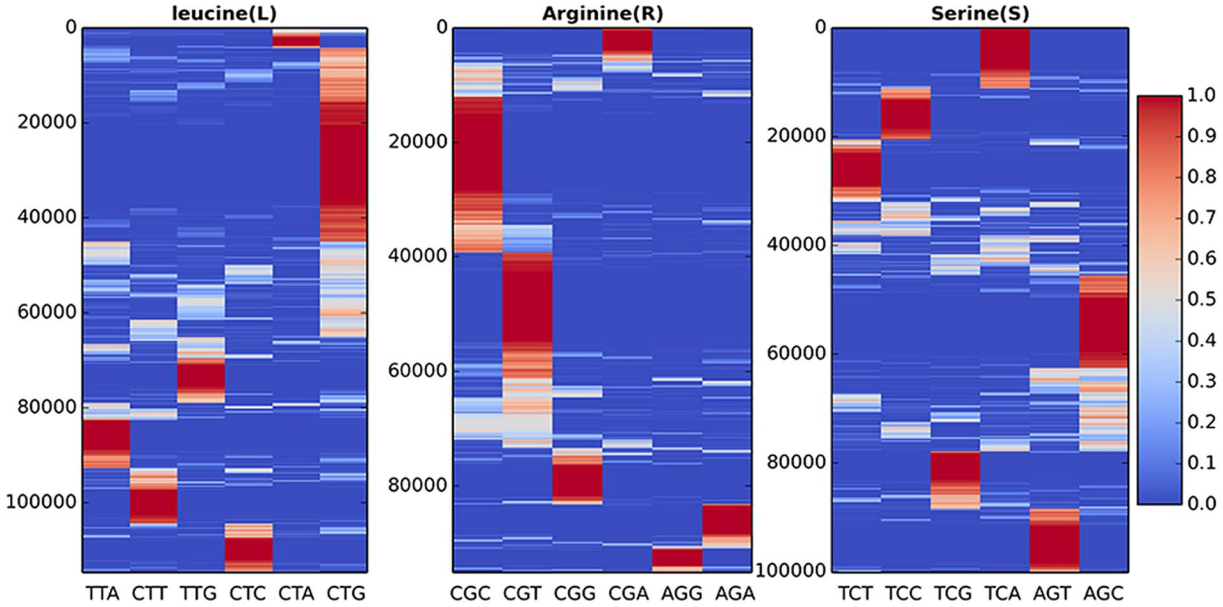

Figure 3. The codon usage patters of Leucine (L), Arginine (R) and Serine (S) in the specific fragment of *E. coli*. All genes of *E. coli* were divided into five-codon windows. The same amino acid fragments were merged and the codon usage bias of the middle amino acid (L, R and S) in the fragment was calculated. Each row represents the codon usage bias of the middle amino acids (L, R and S) in an amino acid fragment with five residues. Each column represents the codons to code the target amino acid. The color from blue to red represents the codon usage frequency of the codon.

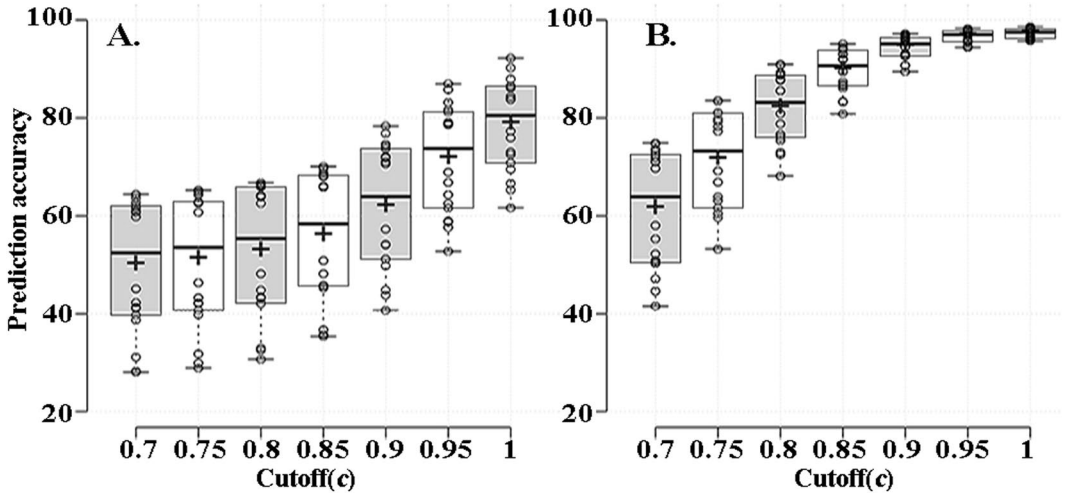

Figure 4. The prediction performance of the 18 classifiers for the 18 amino acids with different matched cutoff and window size (**A**) five amino acids, (**B**) seven amino acids) in *E. coli*. The x-axis is the matched percent and the y-axis is the prediction accuracy of the 18 classifiers. Each open circle represents the prediction accuracy with one of the 18 classifiers. The horizontal divisions (from top to bottom) in each box are the upper whisker, 3rd quartile, median, 1st quartile and lower whisker, respectively. The cross line in each box is the mean prediction accuracy of all 18 classifiers. All of the results were calculated based on a ten-fold cross validation.

a positive contribution to the expression of the reporter genes, indicating that our method could be used to optimize target gene expression without changing the amino acid sequence of the resulting protein.

Discussion

In this study, we proposed a machine learning based method, namely Presyncodon, to design the heterogenous gene with the optimal codon usage for expression. The studies on the two reporter genes (*egfp* and *mApple*) revealed that the low-frequency-usage codons (rare codons) also have the important roles for the functional and soluble expression of the target gene. Here, we also calculated the relation between the strength of relative codon

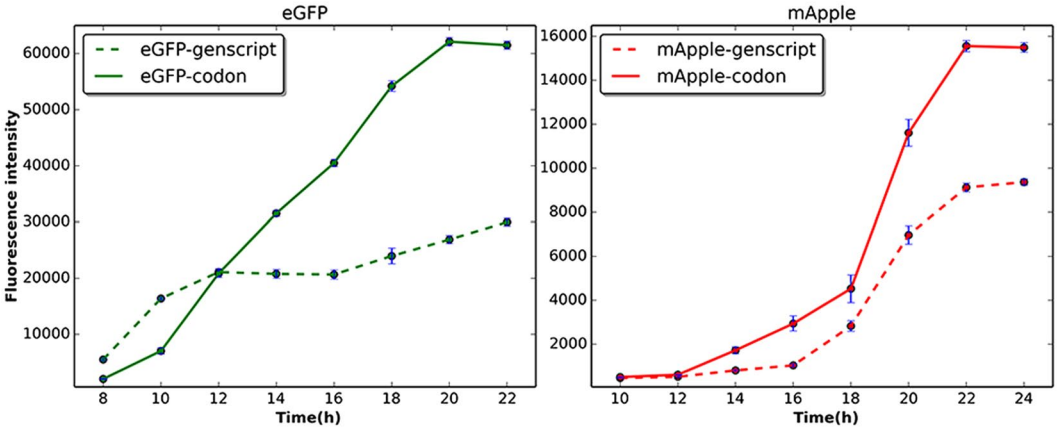

Figure 5. Fluorescence intensity of *E. coli* containing the reporter genes (egfp or mApple). The reporter genes (egfp-codon and mApple-codon) were designed based on the model in this study. The genes (egfp-genscript and mApple-genscript) were designed, in which most of the low-frequency-usage codons were changed to the high-frequency-usage codons of *E. coli* using GenScript software. The strain harboring the corresponding expression plasmid was grown in the auto-induction medium containing 50 μg/mL kanamycin. Data are averages of ten independent experiments. The error bars represent the standard error.

bias (RCBS) and the protein abundance of the *E. coli* genes. As shown in Fig. S11, both the high and low abundant proteins in the cell contains the low- and high-frequency-usage codons. Therefore, the optimal codon for a heterogenous gene should be the appropriate combination between the low- and high-frequency-usage codons. The software Presyncodon was developed to find this interesting combination among synonymous codons for a heterogenous gene.

Therefore, Presyncodon is different from the other software programs such as Gene Designer[26], Gene composer[27] and OPTIMIZER[30] which optimized the heterologous gene with the "preferred" synonymous codons of the expression host. In addition, it was also different to the "codon harmonization" method[35,36]. As a result, Presyncodon did not need to know the information of the codon usage bias of the native host, and it could design any proteins based on the learned knowledge from the *E. coli* genome.

The key of Presyncodon is the codon usage pattern of a specific fragment in the CSI files, but only one residue, which was encouraged by the Google's statistical machine translation (SMT) model. The parameters of the SMT model are derived from the analysis of bilingual text corpora. But the codon usage pattern of a specific fragment was learned from the entire genome database of *E. coli*. Then the predictive models for the expression host *E. coli* were constructed with the learned codon usage patterns. The modes for other important expression hosts, such as *Bacillus subtilis*, and *Pichia pastoris* will be developed in the future.

Materials and Methods

Plasmids and Bacterial Strains. *E. coli* BL21 (DE3) and plasmid pET-30a (+) were used as the expression host and the expression vector for the expression of recombinant proteins. The designed genes (*egfp-codon*, *mApple-codon*, *egfp-genscript* and *mApple-genscript*) were synthesized by GenScript Corporation (Nanjing, China) and inserted into the plasmid pET-30a(+) with the restriction enzymes (*Bam*HI and *Hin*dIII).

Genome Dataset. The dataset of bacterial genomes with full annotation was downloaded from NCBI (ftp://ftp.ncbi.nlm.nih.gov/genomes/Bacteria/) on December 8, 2014. All of the selected subspecies are shown in Table S1.

Phylogenetic Analyses. The 346 genomes in Table S1 were used to carry out the phylogenetic analyses. Those bacterial species had at least five sequenced subspecies except the bacterium *Bacteroides fragilis* NCTC 9343 which was selected as the out-group[38]. The 16 S rDNA sequences of the selected subspecies were collected from the Ribosomal Database Project[40]. All of the 16 S rDNA sequences were aligned with Clustalw[41]. The aligned 16 S rDNA sequences were exported in PHYLIP format for analysis using the PHYLIP set of programs, version 3.696[42]. Similarity matrices of the 16 S rDNA sequences were constructed with the F84 nucleotide substitution model[43,44] using the Dnadist program in PHYLIP. Phylogenetic trees were constructed with the neighbor-joining method with the Neighbor program in PHYLIP. The reliability of the neighbor-joining tree was estimated by bootstrap analysis using 1000 replication datasets generated by the program Seqboot in PHYLIP. For the codon usage trees, 2000 genes were randomly selected from the target genome file, and the Euclidean distances of the codon usage of the subspecies were calculated to construct the corresponding similarity matrices among all of the different subspecies. The 1000 similarity matrices were created by the random selection method. Then, the program Consense in PHYLIP read all of the constructed trees and generated a consensus tree. Trees were drawn and analyzed with the Dendroscope program, version 3.0[45].

Data Clustering and Visualization. All of the data in this study were clustered using the open-source software Cluster, version 3.0[46], and the clustered data were visualized with TreeView, version 1.1.6r4[47].

Construction and Evaluation of the Codon Prediction Model. The *E. coli* genome dataset (Table S1) contains 65 genomes, 64 of which were selected to create a codon selection index (CSI), and one of which (*E. coli* K12_MG1655) was used to evaluate performance of the method. All proteins encoded by the 64 genomes of *E. coli* were split into window sizes of three, five or seven amino acids. The same amino acid fragments from all the genomes were merged into one CSI vector, which contained the codon usage distribution of the middle amino acid and the average codon usage for each amino acid in the fragment. The genome of *E. coli* MG1655 was used to train and evaluate the performance of the model. Every gene in the *E. coli* MG1655 was split into three-, five- or seven-codon windows. Here, the size of the codon window was defined as w. Every short nucleotide sequence was also translated as a short peptide, and then all of the short peptides were searched against the CSI file with the corresponding codon window size. As we wanted to predict the codon selection of the middle amino acid in the peptide, the middle amino acid of the matched peptide and input short peptide must be the same. The matched score (s) between the input peptide and the peptide in the CSI file could be calculated with a BLOSUM62 matrix[48]. The expected maximal score (m) of the input short peptide is the sum of the corresponding diagonal scores in BLOSUM62 matrix of the amino acids in the input peptide. A cutoff (c) could be defined to select the appropriate peptide in the CSI file. Therefore, if the percent (p, $p = s/m$) of the matched score (s) to the expected maximal score (m) is greater than cutoff c, the matched peptide in the CSI file would be selected to update the corresponding input vector. The final input vector of the short peptide is the arithmetic mean of all the possible matched peptides, and the weight of the matched peptide is the matched score(s). Therefore, if an appropriate cutoff c and window size w were defined, every codon in the gene except the first and last w codons could be represented by one vector, and all of the vector were collected as the input dataset to predict the codon selection in *E. coli*. As two (methionine and tryptophan) of the 20 amino acids are coded by only one codon, 18 models for each cutoff c and window size w of the *E. coli* MG1655 genomes were constructed to predict the codon selection with a random forest classifier[49]. The number of trees of the key parameter of the classifier for random forests is 1000. The average overall accuracy and the AUC (Area Under the receiver operating characteristic Curve) for each amino acid model with different parameters were used to evaluate the performance of the method, which was calculated based on a ten-fold cross validation. The AUC was calculated with the package of pROC[50].

Protein Expression and Purification. The reporter genes (*egfp-codon*, *mApple-codon*, *egfp-genscript* and *mApple-genscript*) were cloned into a pET30a(+) expression vector and overexpressed in *E. coli* strain BL21(DE3) pLys. Ten single colonies of the transformed *E. coli* carrying the reporter gene were cultured in liquid Luria-Bertani medium containing 50 μg/mL kanamycin at 30 °C overnight and then inoculated to fresh auto-induction medium (2:100 dilution) and incubated again at 30 °C with shaking at 750 rpm in incubator 1000 (Heidolph, Germany)[51]. The fluorescence intensity was measured at two-hour intervals using a SpectraMax M2 instrument (Molecular Devices, USA). The excitation and emission wavelengths were 484 and 507 nm and 568 and 592 nm, for eGFP and mApple, respectively[52]. The values shown are the averages of ten independent experiments.

Relative codon bias. The strength of relative codon bias (RCBS) was calculated based on the equation in the references[53,54]. The protein abundance data of the *E. coli* was retrieved from paxdB[55].

Method Availability. For non-commercial purposes, the code of the software Presyncodon can be downloaded from http://www.mobioinfor.cn/presyncodon.

Data Availability. All data generated or analyzed during this study are included in this published article (and its Supplementary Information files).

References

1. Pechmann, S. & Frydman, J. Evolutionary conservation of codon optimality reveals hidden signatures of cotranslational folding. *Nature Structural & Molecular Biology* **20**, 237–243, doi:10.1038/nsmb.2466 (2013).
2. Plotkin, J. B. & Kudla, G. Synonymous but not the same: the causes and consequences of codon bias. *Nature Reviews Genetics* **12**, 32–42, doi:10.1038/nrg2899 (2011).
3. Yu, C. H. *et al.* Codon Usage Influences the Local Rate of Translation Elongation to Regulate Co-translational Protein Folding. *Molecular cell* **59**, 744–754, doi:10.1016/j.molcel.2015.07.018 (2015).
4. Li, M. Q. *et al.* Codon-usage-based inhibition of HIV protein synthesis by human schlafen 11. *Nature* **491**, 125–U145, doi:10.1038/nature11433 (2012).
5. Goodman, D. B., Church, G. M. & Kosuri, S. Causes and Effects of N-Terminal Codon Bias in Bacterial Genes. *Science* **342**, 475–479, doi:10.1126/science.1241934 (2013).
6. Hu, S., Wang, M., Cai, G. & He, M. Genetic code-guided protein synthesis and folding in *Escherichia coli*. *The Journal of biological chemistry* **288**, 30855–30861, doi:10.1074/jbc.M113.467977 (2013).
7. Morgunov, A. S. & Babu, M. M. Optimizing membrane-protein biogenesis through nonoptimal-codon usage. *Nat Struct Mol Biol* **21**, 1023–1025, doi:10.1038/nsmb.2926 (2014).
8. Xu, Y. *et al.* Non-optimal codon usage is a mechanism to achieve circadian clock conditionality. *Nature* **495**, 116–120, doi:10.1038/nature11942 (2013).
9. Zhou, M. *et al.* Non-optimal codon usage affects expression, structure and function of clock protein FRQ. *Nature* **495**, 111–115, doi:10.1038/nature11833 (2013).
10. Shalem, O. *et al.* Systematic dissection of the sequence determinants of gene 3′ end mediated expression control. *PLoS Genet* **11**, e1005147, doi:10.1371/journal.pgen.1005147 (2015).
11. Li, G. W., Burkhardt, D., Gross, C. & Weissman, J. S. Quantifying absolute protein synthesis rates reveals principles underlying allocation of cellular resources. *Cell* **157**, 624–635, doi:10.1016/j.cell.2014.02.033 (2014).
12. Thanaraj, T. A. & Argos, P. Protein secondary structural types are differentially coded on messenger RNA. Protein science: a publication of the Protein. *Society* **5**, 1973–1983, doi:10.1002/pro.5560051003 (1996).
13. Saunders, R. & Deane, C. M. Synonymous codon usage influences the local protein structure observed. *Nucleic Acids Research* **38**, 6719–6728, doi:10.1093/nar/gkq495 (2010).

14. Angov, E. Codon usage: Nature's roadmap to expression and folding of proteins. *Biotechnology Journal* **6**, 650–659, doi:10.1002/biot.201000332 (2011).
15. Li, G. W. How do bacteria tune translation efficiency? *Current opinion in microbiology* **24**, 66–71, doi:10.1016/j.mib.2015.01.001 (2015).
16. Fluman, N., Navon, S., Bibi, E. & Pilpel, Y. mRNA-programmed translation pauses in the targeting of *E. coli* membrane proteins. *eLife* **3**, doi:10.7554/eLife.03440 (2014).
17. Kudla, G., Murray, A. W., Tollervey, D. & Plotkin, J. B. Coding-sequence determinants of gene expression in *Escherichia coli*. *Science* **324**, 255–258, doi:10.1126/science.1170160 (2009).
18. Chung, B. K. & Lee, D. Y. Computational codon optimization of synthetic gene for protein expression. *BMC systems biology* **6**, 134, doi:10.1186/1752-0509-6-134 (2012).
19. Coleman, J. R. *et al.* Virus attenuation by genome-scale changes in codon pair bias. *Science* **320**, 1784–1787, doi:10.1126/science.1155761 (2008).
20. Cannarozzi, G. *et al.* A role for codon order in translation dynamics. *Cell* **141**, 355–367, doi:10.1016/j.cell.2010.02.036 (2010).
21. Frenkel-Morgenstern, M. *et al.* Genes adopt non-optimal codon usage to generate cell cycle-dependent oscillations in protein levels. *Molecular Systems Biology* **8**, doi:10.1038/msb.2012.3 (2012).
22. Shah, P. & Gilchrist, M. A. Explaining complex codon usage patterns with selection for translational efficiency, mutation bias, and genetic drift. *Proceedings of the National Academy of Sciences of the United States of America* **108**, 10231–10236, doi:10.1073/pnas.1016719108 (2011).
23. Sharp, P. M., Emery, L. R. & Zeng, K. Forces that influence the evolution of codon bias. *Philosophical Transactions of the Royal Society B-Biological Sciences* **365**, 1203–1212, doi:10.1098/rstb.2009.0305 (2010).
24. Tuller, T., Waldman, Y. Y., Kupiec, M. & Ruppin, E. Translation efficiency is determined by both codon bias and folding energy. *Proceedings of the National Academy of Sciences of the United States of America* **107**, 3645–3650, doi:10.1073/pnas.0909910107 (2010).
25. Chin, J. X., Chung, B. K. & Lee, D. Y. Codon Optimization OnLine (COOL): a web-based multi-objective optimization platform for synthetic gene design. *Bioinformatics* **30**, 2210–2212, doi:10.1093/bioinformatics/btu192 (2014).
26. Villalobos, A. *et al.* Gene Designer: a synthetic biology tool for constructing artificial DNA segments. *BMC Bioinformatics* **7**, 285, doi:10.1186/1471-2105-7-285 (2006).
27. Lorimer, D. *et al.* Gene composer: database software for protein construct design, codon engineering, and gene synthesis. *BMC biotechnology* **9**, 36, doi:10.1186/1472-6750-9-36 (2009).
28. Grote, A. *et al.* JCat: a novel tool to adapt codon usage of a target gene to its potential expression host. *Nucleic Acids Res* **33**, W526–531, doi:10.1093/nar/gki376 (2005).
29. Liu, X., Deng, R., Wang, J. & Wang, X. COStar: a D-star Lite-based dynamic search algorithm for codon optimization. *Journal of theoretical biology* **344**, 19–30, doi:10.1016/j.jtbi.2013.11.022 (2014).
30. Puigbo, P., Guzman, E., Romeu, A. & Garcia-Vallve, S. OPTIMIZER: a web server for optimizing the codon usage of DNA sequences. *Nucleic Acids Res* **35**, W126–131, doi:10.1093/nar/gkm219 (2007).
31. Gustafsson, C., Govindarajan, S. & Minshull, J. Codon bias and heterologous protein expression. *Trends in biotechnology* **22**, 346–353, doi:10.1016/j.tibtech.2004.04.006 (2004).
32. Sorensen, H. P. & Mortensen, K. K. Advanced genetic strategies for recombinant protein expression in *Escherichia coli*. *Journal of biotechnology* **115**, 113–128, doi:10.1016/j.jbiotec.2004.08.004 (2005).
33. Boel, G. *et al.* Codon influence on protein expression in *E. coli* correlates with mRNA levels. *Nature* **529**, 358–363, doi:10.1038/nature16509 (2016).
34. Hurley, J. M. & Dunlap, J. C. CELL BIOLOGY A fable of too much too fast. *Nature* **495**, 57–58 (2013).
35. Angov, E., Hillier, C. J., Kincaid, R. L. & Lyon, J. A. Heterologous protein expression is enhanced by harmonizing the codon usage frequencies of the target gene with those of the expression host. *PLoS One* **3**, e2189, doi:10.1371/journal.pone.0002189 (2008).
36. Buhr, F. *et al.* Synonymous Codons Direct Cotranslational Folding toward Different Protein Conformations. *Molecular cell* **61**, 341–351, doi:10.1016/j.molcel.2016.01.008 (2016).
37. Hillier, C. J. *et al.* Process development and analysis of liver-stage antigen 1, a preerythrocyte-stage protein-based vaccine for Plasmodium falciparum. *Infection and immunity* **73**, 2109–2115, doi:10.1128/IAI.73.4.2109-2115.2005 (2005).
38. Shifman, A., Ninyo, N., Gophna, U. & Snir, S. Phylo SI: a new genome-wide approach for prokaryotic phylogeny. *Nucleic acids research* **42**, 2391–2404, doi:10.1093/nar/gkt1138 (2014).
39. Tuller, T. *et al.* An evolutionarily conserved mechanism for controlling the efficiency of protein translation. *Cell* **141**, 344–354, doi:10.1016/j.cell.2010.03.031 (2010).
40. Cole, J. R. *et al.* Ribosomal Database Project: data and tools for high throughput rRNA analysis. *Nucleic Acids Res* **42**, D633–642, doi:10.1093/nar/gkt1244 (2014).
41. Larkin, M. A. *et al.* Clustal W and Clustal X version 2.0. *Bioinformatics* **23**, 2947–2948, doi:10.1093/bioinformatics/btm404 (2007).
42. Felsenstein, J. PHYLIP - Phylogeny Inference Package (Version 3.2). Cladistics, 164–166 (1989).
43. Felsenstein, J. & Churchill, G. A. A Hidden Markov Model approach to variation among sites in rate of evolution. *Mol Biol Evol* **13**, 93–104 (1996).
44. Kishino, H. & Hasegawa, M. Evaluation of the maximum likelihood estimate of the evolutionary tree topologies from DNA sequence data, and the branching order in hominoidea. *J Mol Evol* **29**, 170–179 (1989).
45. Huson, D. H. & Scornavacca, C. Dendroscope 3: an interactive tool for rooted phylogenetic trees and networks. *Systematic biology* **61**, 1061–1067, doi:10.1093/sysbio/sys062 (2012).
46. de Hoon, M. J., Imoto, S., Nolan, J. & Miyano, S. Open source clustering software. *Bioinformatics* **20**, 1453–1454, doi:10.1093/bioinformatics/bth078 (2004).
47. Saldanha, A. J. Java Treeview–extensible visualization of microarray data. *Bioinformatics* **20**, 3246–3248, doi:10.1093/bioinformatics/bth349 (2004).
48. Styczynski, M. P., Jensen, K. L., Rigoutsos, I. & Stephanopoulos, G. BLOSUM62 miscalculations improve search performance. *Nature biotechnology* **26**, 274–275, doi:10.1038/nbt0308-274 (2008).
49. Breiman, L. Random Forests. *Machine Learning* **45**, 5–32 (2001).
50. Robin, X. *et al.* pROC: an open-source package for R and S+to analyze and compare ROC curv. *es. BMC Bioinformatics* **12**, 77, doi:10.1186/1471-2105-12-77 (2011).
51. Grabski, A., Mehler, M. & Drott, D. The Overnight Express Autoinduction System: High-density cell growth and protein expression while you sleep. *Nat Meth* **2**, 233–235 (2005).
52. Yu, X. *et al.* Identification of a highly efficient stationary phase promoter in Bacillus subtilis. *Scientific reports* **5**, 18405, doi:10.1038/srep18405 (2015).
53. Sabi, R. & Tuller, T. Modelling the efficiency of codon-tRNA interactions based on codon usage bias. *DNA research: an international journal for rapid publication of reports on genes and genomes* **21**, 511–526, doi:10.1093/dnares/dsu017 (2014).
54. Roymondal, U., Das, S. & Sahoo, S. Predicting gene expression level from relative codon usage bias: an application to *Escherichia coli* genome. *DNA research: an international journal for rapid publication of reports on genes and genomes* **16**, 13–30, doi:10.1093/dnares/dsn029 (2009).
55. Wang, M. *et al.* PaxDb, a database of protein abundance averages across all three domains of life. *Molecular & cellular proteomics: MCP* **11**, 492–500, doi:10.1074/mcp.O111.014704 (2012).

Acknowledgements
We thank the Dr. Guanghong Zuo for helping us to carry out the phylogenetic Analyses and Prof. Haiyang Wang for valuable suggestions on this research. We thank Ms. Jaie Woodard for her careful revising of the manuscript and many useful comments. This work was supported by the National High Technology Research and Development Program of China (2013AA102804 to N.W.) and the National Natural Science Foundation of China (NSFC, Grant no. 31371748 to J. T.).

Author Contributions
J.T., N.W. and Y.F. conceived and coordinated the study and wrote the paper. J.T., Y.Y., X.C., and X.L. designed, performed and analyzed the experiments. J.T., Y.Y., Q.Y. and N.W. provided technical assistance and contributed to the preparation of the figures. All authors reviewed the results and approved the final version of the manuscript.

Additional Information
Supplementary information accompanies this paper at doi:10.1038/s41598-017-10546-0

Competing Interests: The authors declare that they have no competing interests.

Publisher's note: Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

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

© The Author(s) 2017