Acinetobacter baumannii

Functional Genome Analysis

BENG 182 Report by
Expected Significance Team:
Troy Sincomb, Teva Bracha, Gayathri Kuppa

Background

Acinetobacter baumannii is a pleomorphic aerobic gram-negative bacillus, commonly found in hospital environments. It is associated with antibiotic resistance and some strains survive in presence of these antibiotics. Scientists studying this opportunistic bacterium observed that specific proteins developed resistance which led to any drug resistant strains. They also observed the mutated genes that develop resistance and were interested to find the function of these proteins.

Our project focused on the functional genome analysis where we took an input file and we would query the sequence using different databases. Some query tools we used were BLAST, Prosite, Kegg, Prody, and Pfam. We basically submitted our protein sequences and scanned them against all the different databases. Our main objective is to get the query results and assign function to certain proteins. This is basically an experimental evolution project where we row an organism in a culture, and give it a certain antibiotic. We then observe if the organism can survive in the presence of these antibiotics. It is important to see which genes are mutated and which combination of drugs can be given even with the antibiotic resistance. We can also note which proteins, developed by the gene, survive in the presence of antibiotics and which gene confer to resistance.

We focused on sequences 301-400 to annotate the of function assignment on proteins as well as assign functional analysis to the genome as a whole. After we query sequences using different databases, we extract the functions assignments into a table. Each entry consists of keywords that are automatically extracted from the known annotations.

Methods

A pipeline was made to annotate the sequences 301 to 400. This pipeline consisted of: Biopython Blast, Prody module, Biopython ScanProsite, Bioservices Kegg, pfam2go & prosite2go geneontologies. All the databases in use were queried via the respective protein sequence except for Kegg. We were unable to find a usable id from the

Blast output and instead used the gene id from the original annotations [3]. Biopython Blast query results were parsed from the same module to get the top 5 hits that had an expected value of at least 1e-20 [1]. Prosite was filtered to only present the top 10 hits. Pfam data was queried using Prody and filtering was not necessary because there were often only one hit [2]. Lastly, using the pfam and prosite ids, the go-slim geneontologies for Pfam and Prosite were downloaded and parsed. We decided to use both geneontologies in case the protein sequences had only outputs from just one database.

The overall runtime was on average 6.5 hours. This is predominantly due to the using the traditional Blastp "nr" and having their webserver only allowing one IP address at a time. All other database queries were just about a few seconds a sequence. If the Blast database were downloaded (about 80 GBs), the runtime would be down to minutes. For extra credit, sequences 801 to 900 were also put into the pipeline used for sequences 301 to 400.

Example of the 301 to 400 annotations in Pandas:

	id	blast	pfam	prosite	kegg	go	comments
0	ABO11628	[['gi|1196949756|ref|WP_086377352.1|', 'nuclea...	noHomolog	noHomolog	noHomolog	noHomolog	301 -> nuclease; lamin tail domain -> Likely t...
1	ABO11629	[['gi|446703992|ref|WP_000781338.1|', 'MULTISP...	{'PF02798': {'accession': 'PF02798.19', 'class...	[{'sequence_ac': 'USERSEQ1', 'start': 1, 'stop...	{'acb00480': 'Glutathione metabolism'}	['GO:protein binding ; GO:0005515']	302 -> glutathione S-transferase -> Enzyme tha...
2	ABO11630	[['gi|1098612794|ref|WP_071243622.1|', 'alkyl ...	{'PF07992': {'accession': 'PF07992.13', 'class...	[{'sequence_ac': 'USERSEQ1', 'start': 97, 'sto...	noHomolog	['GO:oxidoreductase activity ; GO:0016491', 'G...	303 -> alkyl hydroperoxide reductase subunit F...
3	ABO11631	[['gi|126387133|gb|ABO11631.1|', 'alkyl hydrop...	noHomolog	[{'sequence_ac': 'USERSEQ1', 'start': 115, 'st...	noHomolog	noHomolog	304 -> alkyl hydroperoxide reductase subunit F...
4	ABO11632	[['gi|445980391|ref|WP_000058246.1|', 'MULTISP...	{'PF07866': {'accession': 'PF07866.10', 'class...	noHomolog	noHomolog	noHomolog	305 -> Hypothetical Protein -> In the DUF1653 ...
5	ABO11634	[['gi|691014800|ref|WP_031976756.1|', 'lysine ...	{'PF13619': {'accession': 'PF13619.5', 'class'...	noHomolog	noHomolog	noHomolog	306 -> lysine tRNA synthetase; KTSC domain pro...
6	ABO11636	[['gi|126387138|gb|ABO11636.1|', 'ATP-dependen...	{'PF11898': {'accession': 'PF11898.7',	[{'sequence_ac': 'USERSEQ1',	noHomolog	noHomolog	307 -> ATP-dependent (RNA) helicase -> Uses

Screen shot of initial pipeline:

```
def blast(query):
    try:
        result_handle = NCBIWWW.qblast("blastp", "nr", query.seq)
        result_handle = result_handle.read()
        return result_handle
    except:
        return 'noHomology'

def pfam(seq):
    try:
        matches = searchPfam(seq)
        return matches
    except:
        return 'noHomology'

def prosite(seq):
    try:
        handle = ScanProsite.scan(seq=seq, lowscore=10) #add lowscore=# to get the next # 'possible'
          additional hits
        result = ScanProsite.read(handle)
        if result == []:
            return 'noHomology'
        else:
            return result
    except:
        return 'noHomology'

def kegg(gn):
    try:
        hit = k.get_pathway_by_gene(gn, "acb")
        if not hit:
            return 'noHomology'
        else:
            return hit
    except:
        return 'noHomology'
```

Screen shot of Blast Parsing:

```
def blast_parser(blast):
    fields = []
    result_handle = StringIO(blast)
    blast_records = NCBIXML.parse(result_handle)
    for rec in blast_records:
        for alignment in rec.alignments:
            for hsp in alignment.hsps:
                if hsp.expect < 1e-20:
                    fields.append([alignment.hit_id, alignment.hit_def, alignment.accession, str(hsp.expect)])
            if len(fields) >= 5: break
    result_handle.close()
    return fields
```

Commenting Process

Once all the entries were obtained, they were commented using a combination of our BLAST, Pfam, Prosite, and KEGG results. Our BLAST results were our primary source for determining function. We cross-checked the function assigned by BLAST with Pfam and KEGG to confirm the functions and domains matched properly, then did further research to determine the function of the protein in the context of the cell. In cases with hypothetical proteins, we checked Pfam and

Prosite to see if we could learn anything about the domains within. Comments were filled in indicating any Pfam families they were involved in, and proteins which were hypothetical and had no matches in the other databases were left blank. Although 82 sequences would be commented and properly annotated, 18 sequences could not. Those 18 sequences are: 305, 312, 315, 316, 318, 319, 338, 343, 367, 371, 377, 379, 380, 381, 385, 389, 394, and 396.

Antibacterial Resistance Genes

We found five genes that seemed most likely to cause antibacterial resistance.

The first was a metallo-beta-lactamase hydrolase. This protein causes antibiotic resistance by breaking down beta-lactams into beta-amino acids. Once the beta-lactams lose their structure, they can no longer inhibit peptidases by mimicking the structure of D-Ala-D-Ala, which is normally a building block of peptidoglycan, causing cell wall synthesis to stop.

The second protein was a multiple antibiotic resistance regulator or MarR proteins. This protein functions by repressing gene expression of drug efflux proteins when various antibiotics and toxins are absent, and conversely activating the transcription of these proteins when they are present. The efflux proteins function by pumping the antibiotic out of the cell before it can do any harm, thereby neutralizing it.

The third set of proteins were Tetracycline regulators or TetR proteins. These proteins function similarly to MarR in that they repress the transcription of TetA when tetracycline is absent. Tetracycline functions by binding to the 30S ribosomal subunit found in the cytoplasm. The TetA protein pumps tetracycline out of the cell before it can bind to the 30S ribosomal subunits neutralizing its effect.

The fourth protein was a part of peptidase M15 family. These peptidases have the potential to be VanX or VanA, peptidases which

contribute to antibiotic resistance from glycopeptides. VanX functions by breaking down D-Ala-D-Ala in the cell causing it to not be used in the formation of the peptidoglycan of the cell wall. VanA creates D-Ala-D-Lac which can be used in place of D-Ala-D-Ala to construct peptidoglycan and is importantly unaffected by the actions of Vancomycin or Teicoplanin.

Conclusion

By assigning functionality to the protein sequences, we were able to detect key genes that relate directly to antibacterial resistance. Genome sequences have served as the foundation for the advancement in investigating protein functionality. Genome sequences are important in determining complex cellular functions. Genome sequencing projects also offer many insights in determining new treatments for various diseases. Genome sequences have served as the foundation for the advancement in investigating protein functionality.

References

1. Cock PA, Antao T, Chang JT, Chapman BA, Cox CJ, Dalke A, Friedberg I, Hamelryck T, Kauff F, Wilczynski B and de Hoon MJL (2009) Biopython: freely available Python tools for computational molecular biology and bioinformatics. *Bioinformatics*, 25, 1422-1423
2. Bakan A, Meireles LM, Bahar I. ProDy: Protein Dynamics Inferred from Theory and Experiments. *Bioinformatics* **2011** 27(11):1575-1577.
3. Thomas Cokelaer, Dennis Pultz, Lea M. Harder, Jordi Serra-Musach, Julio Saez-Rodriguez; BioServices: a common Python package to access biological Web Services programmatically. *Bioinformatics* 2013; 29 (24): 3241-3242. doi: 10.1093/bioinformatics/btt547
4. Carpenter, A.E. & Sabatini, D.M. Systematic genome-wide screens of gene function. Nat. Rev. Genet.5, 11–22 (2004)

5. Gerlt, J.A. & Raushel, F.M. Evolution of function in (beta/alpha)8-barrel enzymes. Curr. Opin. Chem. Biol.7, 252–264 (2003)