Locating and identifying splicing elements near the optional 6th exon in the D2 receptor gene

Heinz, D. A.[1,2] , Ritz, A[1]

[1]*Reed College Computational Biology, Portland, Oregon, USA*

[2] *OHSU Vollum Institute, Portland, Oregon, USA*

Data and data formatting

This program requires three separate pieces of data: a gene sequence, the splice sites of that gene, and a database of splicing motifs.

The gene sequence must be in fasta form, and in the same folder as the python folder. Its file name will be used as an argument of several functions. In this case, the gene is that of the D2 receptor in *Mus musculus* (Eppig et al., 2015).

The program is designed to work with a specific data set from SpliceAid-F (Giulietti, M et al. 2012). In this case, the data set is saved as spliceData.txt. No modifications were made to this data set.

Finely, a list of splice sites are easily obtained from the genbank entry on a given gene. They must be transcribed from the genbank entry into a list and that list will be an argument in several of the functions in the program.

High level steps

This program can easily be broken up into four distinct steps: reading and formatting data, motif finding, splitting by splice site, and category look up.

For reading and formatting data, the gene fasta is read and saved as a string. The splicing data set is read into a list of lists, with each item in the inner lists representing a category, and each item in the outer list representing a distinct motif. Unfortunately, there does not seem to be a good data set for splice sites by gene, so thee splice site list must be manually copied down from the genbank entry on the gene of interest.

The motif finding step is simple: given a list of motifs and a gene, it will create a dictionary with each motif found within the gene as a key, and a list of the locations of the motif as the value associated with each motif. It can also take distance as a function, allowing it to call a motifs in the gene with up to ‘distance’ differences from the splice motif as a hit.

The splicing by splice site step not only breaks the gene into sections of a given size around a given set of splice sites, but also builds a dictionary with the keys as each splice site, and the values the output of the motif finding step for the section of the gene with the given range of the specific splice site.

The category lookup step builds a dictionary with the keys as each motif for a given species and the values as another dictionary with the keys as the categories given in the splice dataset and the values as the associated value for that category and that motif. Another function then can, given a motif, a species, and a category of interest, output the associated value for that motif and that category.

Results and Discussion

The process of splicing a gene in a specific way is highly complex, and a computational approach to determining what determines splice sites, while potentially very useful, cannot give the entire picture. A motif that in one context acts as a splicing element in another context will not, and one splicing factor may bind to two different splicing elements and have the same action. Additionally, combinatorial action of multiple splicing factors may have non-additive and difficult-to-predict effects on splicing. Given this, any 'results' observed in this project must be seen not as an answer to the question, but as an indication of what molecular biologists should see as candidates for investigation.

I set out to determine which splicing motifs could be involved in the differential splicing of the D2 long form versus short form (short form is missing exon 6). That my results do not shed light on this question is not particularly surprising, as it is likely to be a complicated process, and there are many assumptions that would have to be made to draw any conclusions (what hamming distance is acceptable, what distance from the splice site is acceptable, that we do in fact have data on the particular splicing element which is active in this case, etc.).

That being said, this project has yielded at least one potentially interesting result: motifs which have been implicated in the splicing of the ionic AMPA glutamate receptor (GluR2) gene (gria2) (Eppig et al., 2015) occur at far higher rates than is probable within 300 bp of each splice site in the D2 receptor gene. This gene also undergoes alternative splicing in a similar manner to the D2 gene. This is potentially really, really cool, neurologically speaking.

D2 alternative splicing in midbrain dopamine neurons seems to be part of the cocaine tolerance pathway, in which alternate splices affect the trafficking of the receptor to the surface of the cell. The D2 receptor is inhibitory when interacting with dopamine, and is part of a negative feedback loop in which midbrain

dopamine neurons, when stimulated, self-innervate via dendritic release of dopamine, producing an inhibitory current leading to more required activating input in the near future.

Differential expression of the GluR2 receptor also seems to be implicated in the midbrain cocaine tolerance pathway, although the mechanism of its involvement is less clear (Tang, Fasulo, Mash, & Hemby, 2003). This provides an interesting context for the observation that the D2 and the GluR2 genes seem to share specific splicing motifs. It is possible that this is a random happenstance, but I believe, based on the number of coincidences which would have had to occur, that this is indicative of something involving a shared mechanism. The working hypothesis I would go forward with would be that there is a splicing factor or a set of several splicing factors which mediate the cocaine tolerance pathway through action on at least the D2 and GluR2. Future experiments should attempt to elucidate this relationship (thesis?).

Generalizability

I consider myself very lucky to have found anything interesting with this project. My goal was to create the most generalizable algorithm I could within the parameters of this problem. This specific application happens to work quite well, but the program would work with very little tampering for any task in which a researcher wanted to look for approximate motifs of any length (or type… we could us this for analyzing protein domains or even literature, if we wanted to) in any large sequence. The specificity of the locations and ranges for searching could make this less brute force, and could be used to look for, for example, TEs near particularly unstable areas of the genome, or a set of mutations to repressors near oncogenes, or all the context of all verbs in a variety of texts in any given language (this relates closely to my girlfriend's linguistic thesis). The list could go on. This is, of course, not a new ability. Motif finders are not difficult to write, and many of them exist. Because I wrote all of the code from scratch and so understand it very well, however, I have the freedom to modify the code to make it perfect for any of these uses. To me, this is very powerful, and a tool I look forward to exploiting throughout my scientific career.

References

Eppig JT, Blake JA, Bult CJ, Kadin JA, Richardson JE;; The Mouse Genome Database Group. 2015. The Mouse Genome Database (MGD): facilitating mouse as a model for human biology and disease. Nucleic Acids Res. 2015 Jan 28;43(Database issue):D726-36.

GIULIETTI, M et al. SpliceAid-F: a database of human splicing factors and their RNA binding sites. **EMBnet.journal**, [S.l.], v. 18, p. pp. 73-74, apr. 2012. ISSN 2226-6089. Available at: <http://journal.embnet.org/index.php/embnetjournal/article/view/423/608>. Date accessed: 13 Dec. 2015. doi:http://dx.doi.org/10.14806/ej.18.A.423.

Tang, W. X., Fasulo, W. H., Mash, D. C., & Hemby, S. E. (2003). Molecular profiling of midbrain dopamine regions in cocaine overdose victims. *Journal of Neurochemistry, 85*(4), 911-924. doi:10.1046/j.1471-4159.2003.01740.x