BT5240: Computer Simulations of Biomolecular Systems

Final Course Project

Submitted by Sahana Gangadharan (BE17B038)

The paper I've chosen for the project is -

De Oliveira CCS, Pereira GRC, De Alcantara JYS, Antunes D, Caffarena ER, De Mesquita JF. **In silico analysis of the V66M variant of human BDNF in psychiatric disorders: An approach to precision medicine**. *PLoS One*. 2019;14(4):e0215508. Published 2019 Apr 18. doi:10.1371/journal.pone.0215508

I The objective of the paper you have chosen for the project -

Brain-derived neurotrophic factor (BDNF) plays a very important role in neurogenesis and synapse formation. The mutated form of this protein - V66M is the most prevalent BDNF mutation in humans which impairs the function and distribution of BDNF and causes various psychiatric disorders. The pro-region of BDNF, particularly position 66 and its adjacent residues, are determinant for the intracellular sorting and activity-dependent secretion of BDNF. Hence the key objective of this paper is to study the molecular dynamics of V66M mutated protein for understanding its effect on psychiatric disorders.

II The reason I chose this article -

My interest in neuroscience and how the brain functions started very early on in my academic journey. This article explains the dynamics involved in how a point mutation in a neurotrophic factor would result in psychiatric disorders. Further, the molecular analysis of the dynamics of the wild type protein is compared with the dynamics of the mutated protein, thereby showing the comparative behaviour of both the proteins.

This paper also required the generation of a complete theoretical 3D model for the protein and its corresponding mutation. The exposure to new softwares is always thrilling and exciting, especially if the software can predict biological features with some precision. In this article, the protein is put in a PBC box that is not of the conventional cubic shape, but is of a dodecahedron shape. This was something intriguing and I wanted to take this as an opportunity to analyse biomolecules with different constraints.

Finally, the MD simulations that are required for analysing the two biomolecules are very similar to what was given for practice in the tutorial with small differences in the details. Therefore, the project was a great opportunity to re-learn the basics of MD simulations again with more straightforward analysis to perform.

III Describe what results you want to reproduce and why?

Trajectory analysis was performed on BDNF_WT (wildtype protein) and BDNF_V66M (mutated protein). In the paper, Root-mean-square deviation (RMSD), root-mean-square fluctuation (RMSF), B-factor, radius of gyration (Rg), solvent accessible surface area (SASA), and secondary structure were calculated separately for each triplicate trajectory, taking the initial structure of the production dynamics as the reference, to investigate biochemical and structural changes of the native and mutant structures.

Before analysing these functions, and immediately after MD production simulations, it is necessary to remove the PBCbox from the trajectory, to avoid erroneous results when molecules of the protein move into the spaces of adjacent boxes. This is performed using the gmx trjconv command.

```
gmx trjconv -f md.xtc -o md_nopbc.xtc -s md.tpr -pbc mol -ur compact
```

- -f md_nopbc.xtc has the trajectory of the protein at all time frames recorded at 10 ps each (as specified in the paper).
- -o md_nopbc.xtc is the output trajectory file without the PBC box.
- -s md.tpr provides the structure file for the reference structure.
- -pbc mol sets the periodic boundary condition treatment; mol puts the centre of mass of molecules.
- -ur compact sets the unit cell representation; compact puts all atoms at the closest distance from the center of the box.

Root mean square deviation (RMSD)

The RMSD function provides the euclidean distance between every backbone atom provided in two GROMACS structures. Each structure that is recorded during the MD simulation (for 1 ns) is compared with a reference structure (1st frame) specified in the command. RMSD is therefore useful for analyzing the time-dependent motion of a given structure and for determining its spatial convergence throughout the simulation. While running the command, select backbone atoms to compute the Least square fit and Protein for calculating the RMSD function.

The command for calculating RMSD is as follows -

```
gmx rms -f md_nopbc.xtc -s md.tpr -o rmsd.xvg
```

- -f md_nopbc.xtc has the trajectory of the protein at all time frames recorded at 10 ps each (as specified in the paper).
- -s md.tpr provides the structure file for the reference structure, against which RMSD is calculated.
- -o rmsd.xvg is where the output file is stored in a .xvg format.
- Choose Backbone (4) for calculating the Least Square Fit and Protein (1) for computing RMSD values.

Root mean square fluctuation (RMSF)

While RMSD is calculated for the whole protein at every frame of the trajectory, the RMSF function is calculated for each residue to identify the measure of local flexibility, thermal stability and heterogeneity of macromolecules based on displacement of each residue. It is calculated for the C-alpha atom (3 in GROMACS) of each residue by taking the square root of the variance of the fluctuation around the average position.

The command for calculating RMSF is as follows. The specifications are similar to that mentioned for RMSD. -

```
gmx rmsf -f md_nopbc.xtc -s md.tpr -o rmsf.xvg
```

Radius of Gyration (Rg)

The Rg function measures the compactness and the dimensions of the protein structure. The command in GROMACS gives the length of the protein or in other words, the radial measure of the protein from the centre of mass (COM). While running the command, select Protein (1) to compute Rg (every atom in the protein is taken into account).

The command for calculating Rg is as follows -

```
gmx gyrate -f md_nopbc.xtc -s md.tpr -o gyrate.xvg
```

Solvent accessible surface area (SASA)

SASA is defined as the exposed surface of a protein that can be accessed by a solvent. Thus, SASA analysis is helpful in understanding the protein's ability to interact with solvents and other molecules. It also gives insights about whether the protein's overall structure is moving from an open to a close state or vice versa. While running the command, select Protein (1) to compute SASA (every protein atom, including the hydrogen atoms are taken into account for calculating the accessible surface area).

The command for calculating SASA is as follows -

```
gmx gyrate -f md_nopbc.xtc -s md.tpr -o area.xvg
```

These are the four analyses that will be performed as a part of this project. These four functions were chosen for the ease of replication and the direct analysis that follows.

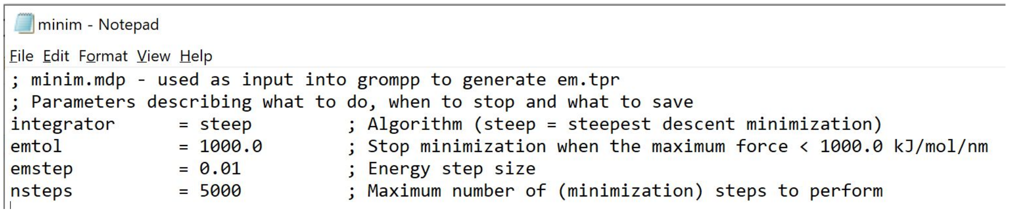
IV Provide the simulation details as given in the article

- The force field chosen for the simulations - AMBER99SB-ILDN.
- Water model for solvation - TIP3P water model.
- PBC box details - The PBC box is a dodecahedron box with the dimensions of 5.50 *8.27*4.56 nm.
- The salt concentration of the system is set to 0.15 mol/l in the neutralization step.
- The net charge observed in the protein (and confirmed using BetChem?) is +7. Hence, to neutralize the system, Na+ and Cl- ions were added accordingly which resulted in the addition of 19 Na+ ions and 26 Cl- ions.
- The minimization step was performed using the steepest descent method with a total of 5000 steps (The system was however minimised in 587 steps itself).
- Following minimization, NVT, NPT and the production steps were performed. NPT simulation was done at 300 K and 1 atm pressure, for a duration of 100 ps. Bonds were constrained using the posre.itp file in both the NVT and NPT simulations.
- For all the MD simulations, Parrinello-Rahman is chosen as a barostat and v-rescale is chosen as the thermostat, both of which lasted for 100 ps.'
- The MD production simulation is performed at 300 K for 200 ns (1 ns for this project) for both the WT and the V66M variant. The LINCS (linear constraint solver) algorithm is applied to constrain covalent bonds and electrostatic interactions are processed using the particle mesh Ewald (PME) method. The MD trajectories are recorded at every 10 ps.

V Have you modified the *.mdp files used in your Assignment II? If so, enlist or highlight the modifications with justifications

ions.mdp - There are no changes done to the ions.mdp file.

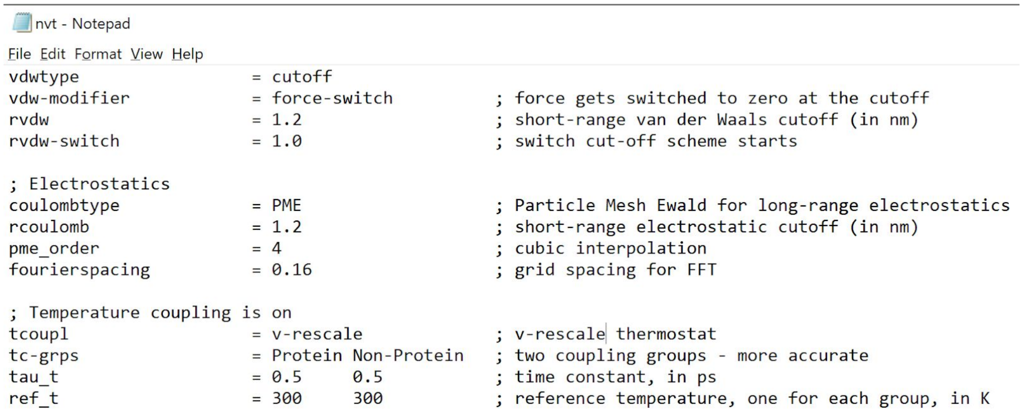
minim.mdp - The number of steps for minimization is changed from 50,000 to 5000 as mentioned in the article.



```
; minim.mdp - used as input into grompp to generate em.tpr
; Parameters describing what to do, when to stop and what to save
integrator      = steep         ; Algorithm (steep = steepest descent minimization)
emtol           = 1000.0        ; Stop minimization when the maximum force < 1000.0 kJ/mol/nm
emstep          = 0.01          ; Energy step size
nsteps          = 5000          ; Maximum number of (minimization) steps to perform
```

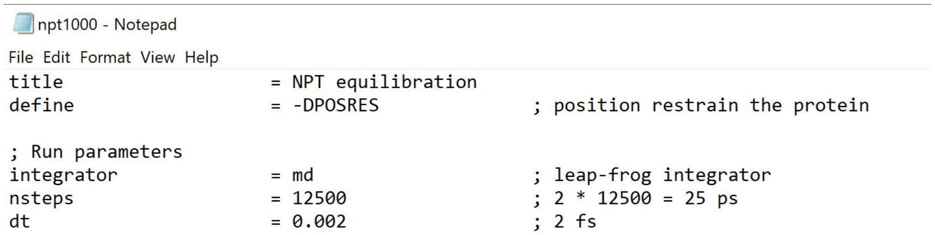
This change barely affects any further process, since the minimization process is stopped in about 587 steps itself. Hence, having a smaller number (5000 < 50,000) for nsteps does not affect the simulation much.

nvt.mdp - The thermostat is changed from Nose-Hoover thermostat to **v-rescale thermostat** as specified in the paper. V-rescale (velocity rescaling) is a constraint method which is comparatively good for starting runs in obscure cases as loss of heat can be transferred at each time-step. The velocities of the particles are scaled by a factor of λ for T(t) to reach T^o. V-rescale is useful as it is a good way to input or remove thermal energy in a quick fashion and it ensures that the simulation doesn’t crash. V-rescale is usually enforced at the beginning of simulations so that it can handle the energy fluctuations better than the other available thermostats. In this case, because the PDB file for the whole protein is unavailable, it is considered safe to start the simulations with v-rescale instead of the established nose-hoover thermostat.

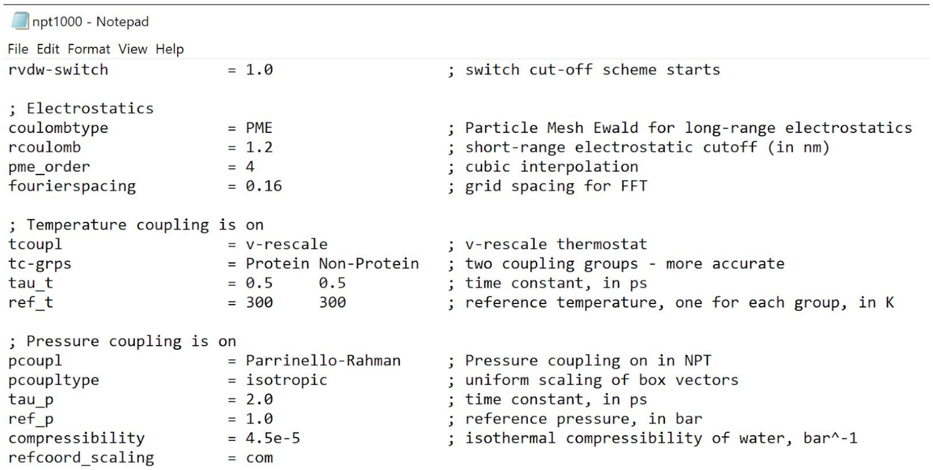


```
vdwtype                 = cutoff
vdw-modifier            = force-switch          ; force gets switched to zero at the cutoff
rvdw                    = 1.2                   ; short-range van der Waals cutoff (in nm)
rvdw-switch             = 1.0                   ; switch cut-off scheme starts

; Electrostatics
coulombtype             = PME                   ; Particle Mesh Ewald for long-range electrostatics
rcoulomb                = 1.2                   ; short-range electrostatic cutoff (in nm)
pme_order               = 4                     ; cubic interpolation
fourierspacing          = 0.16                  ; grid spacing for FFT

; Temperature coupling is on
tcoupl                  = v-rescale             ; v-rescale thermostat
tc-grps                 = Protein Non-Protein   ; two coupling groups - more accurate
tau_t                   = 0.5     0.5           ; time constant, in ps
ref_t                   = 300     300           ; reference temperature, one for each group, in K
```

npt.mdp - The NPT simulation was broken into 4 stages where the force constant values in posre.itp is brought down from 1000 to 500 to 250 to 0 gradually and equally spaced in 100 ps. Further, since 50,000 steps were carried out, 12,500 steps were performed in each stage for 25 ps.



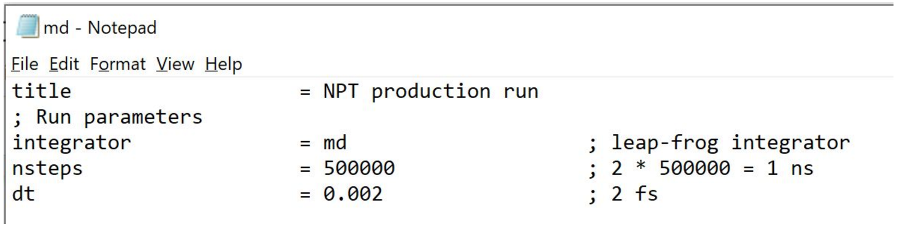
```
title                   = NPT equilibration
define                  = -DPOSRES              ; position restrain the protein

; Run parameters
integrator              = md                    ; leap-frog integrator
nsteps                  = 12500                 ; 2 * 12500 = 25 ps
dt                      = 0.002                 ; 2 fs
```



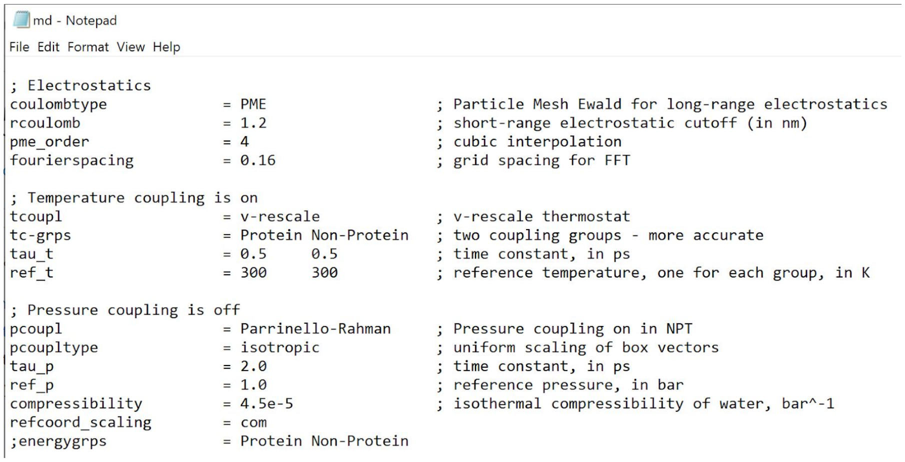
```
rvdw-switch             = 1.0                   ; switch cut-off scheme starts

; Electrostatics
coulombtype             = PME                   ; Particle Mesh Ewald for long-range electrostatics
rcoulomb                = 1.2                   ; short-range electrostatic cutoff (in nm)
pme_order               = 4                     ; cubic interpolation
fourierspacing          = 0.16                  ; grid spacing for FFT

; Temperature coupling is on
tcoupl                  = v-rescale             ; v-rescale thermostat
tc-grps                 = Protein Non-Protein   ; two coupling groups - more accurate
tau_t                   = 0.5     0.5           ; time constant, in ps
ref_t                   = 300     300           ; reference temperature, one for each group, in K

; Pressure coupling is on
pcoupl                  = Parrinello-Rahman     ; Pressure coupling on in NPT
pcoupltype              = isotropic             ; uniform scaling of box vectors
tau_p                   = 2.0                   ; time constant, in ps
ref_p                   = 1.0                   ; reference pressure, in bar
compressibility         = 4.5e-5                ; isothermal compressibility of water, bar^-1
refcoord_scaling        = com
```

The Thermostat was again changed to v-rescale, for the above reasons. The above changes (nsteps and tcoupl) were incorporated in npt1000.mdp, npt500.mdp, npt250.mdp and npt0.mdp files.

md.mdp - The thermostat was changed to v-rescale for the same reason and the total time for MD production was restrained to 1 ns and not 200 ns as mentioned in the article.



```
title                   = NPT production run
; Run parameters
integrator              = md                    ; leap-frog integrator
nsteps                  = 500000                ; 2 * 500000 = 1 ns
dt                      = 0.002                 ; 2 fs
```



```
; Electrostatics
coulombtype             = PME                   ; Particle Mesh Ewald for long-range electrostatics
rcoulomb                = 1.2                   ; short-range electrostatic cutoff (in nm)
pme_order               = 4                     ; cubic interpolation
fourierspacing          = 0.16                  ; grid spacing for FFT

; Temperature coupling is on
tcoupl                  = v-rescale             ; v-rescale thermostat
tc-grps                 = Protein Non-Protein   ; two coupling groups - more accurate
tau_t                   = 0.5     0.5           ; time constant, in ps
ref_t                   = 300     300           ; reference temperature, one for each group, in K

; Pressure coupling is off
pcoupl                  = Parrinello-Rahman     ; Pressure coupling on in NPT
pcoupltype              = isotropic             ; uniform scaling of box vectors
tau_p                   = 2.0                   ; time constant, in ps
ref_p                   = 1.0                   ; reference pressure, in bar
compressibility         = 4.5e-5                ; isothermal compressibility of water, bar^-1
refcoord_scaling        = com
;energygrps             = Protein Non-Protein
```

VI Create a VMD movie of your final 1 nm production run, include only frames for every 100 ps

The video and the screenshots for both the wildtype and the V66M variant of BDNF protein can be found here.

VII Complete description of how you have computed the properties of interest

The following are the steps/commands required to set up the system and run the simulation.

Step 1 - Generate the theoretical structure of BDNF_WT using the Rosetta server.

The article uses a few structure prediction tools to identify the 3D structure of BDNF and finally reports that the structure predicted using the Rosetta server had the lowest RMSD with PDB:1BND[1] The sequence of BDNF from UNIPROT is first given as an input to the Rosetta[2] server and the output model is checked with TM-Align server for RMSD < 2.0 A° and TM-Scores approaching 1.

My output result (BDNF.pdb) had a RMSD of 1.07 A° and TM-Score of 0.93933. Although not the exact result as that mentioned in the paper, this result seemed accurate enough to proceed forward.

Validation of the structure was performed using Verify-3D, RAMPAGE and QMEAN. The figures were all congruent with the ones mentioned in the article, hence the validity of BDNF.pdb is proved.

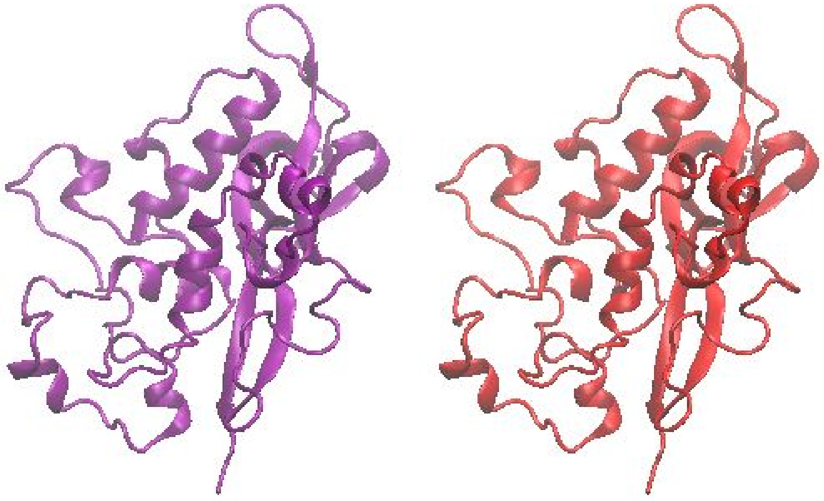


Figure1 *(i) BDNF.gro - Purple color & NewCartoon style, (ii) BDNF_ V66M.gro - Red color & NewCartoon style*

Step 2 - Generate the mutation (V66M_BDNF.pdb) using the *Mutator Plugin* tool in the VMD application. This will require the generation of BDNF_autopsf.psf file before the PDB file for the V66M variant. Analyse in VMD application to observe the similarity in structure, between the WT and the V66M variant.

Note - Steps 3,4,5,6 are done for both the WT and the V66M variants.

Step 3 - Compare all the simulation details as given in the article, with the the *.mdp files present in your system and make the necessary amendments.

Step 4 - The preliminary steps involve solvation and neutralization of the system. There will be a small discrepancy regarding the number of Na+ and Cl- ions that are added between the article and the solution. In this project, the concentration of solvent (0.15 mol/L) is maintained and the required ions are added.

[1] The structure of a short segment of BDNF is deposited to PDB as 1BND.

[2] Now termed Robetta - https://robetta.bakerlab.org/submit.php

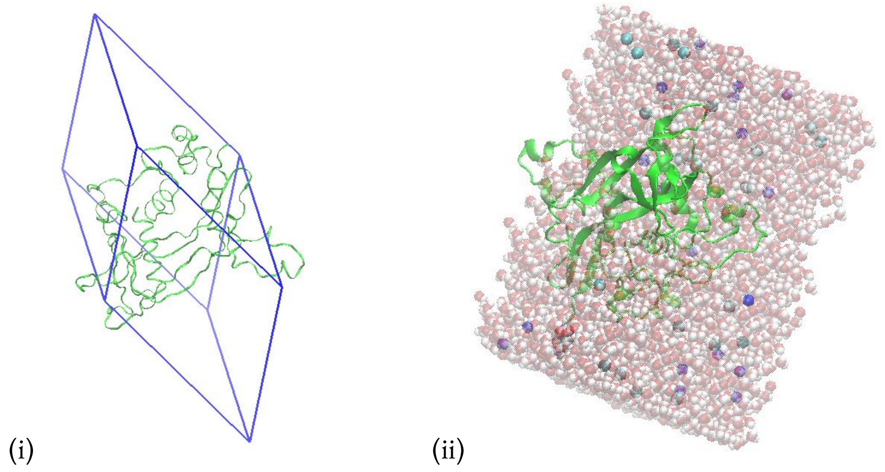

(i) (ii)

Figure2 *(i) BDNF protein in a dodecahedron box (ii) BDNF protein after solvation and neutralization*

Step 5 - MD simulations - NVT simulation is carried out for 100 ps with the details specified earlier. NPT simulation is performed in four stages and, in between each step, the force constant in the posre.itp file is altered. MD production is performed without any position restraint for 1 ns.

Step 6 - MD Analysis - Before proceeding with the analysis, it is necessary to remove the PBC box off the trajectory. Presence of a PBC box might give erroneous results when certain molecules move during the MD simulations. In GROMACS functions such as Radius of gyration, Root Mean Square Deviation (RMSD), Root mean Square Fluctuation (RMSF) and Solvent Accessible Surface Area (SASA) are analysed. The commands for analysis are fairly simple and they are listed above.

Step 7 - Plotting graphs for analysis - The *.XVG files generated by GROMACS are imported into MATLAB for plotting the graphs. The codes for analysis are attached here.

VIII Compare your results with that reported in the article. If there are any discrepancies explain it with reasons.

Root Mean Square Deviation (RMSD) Analysis -

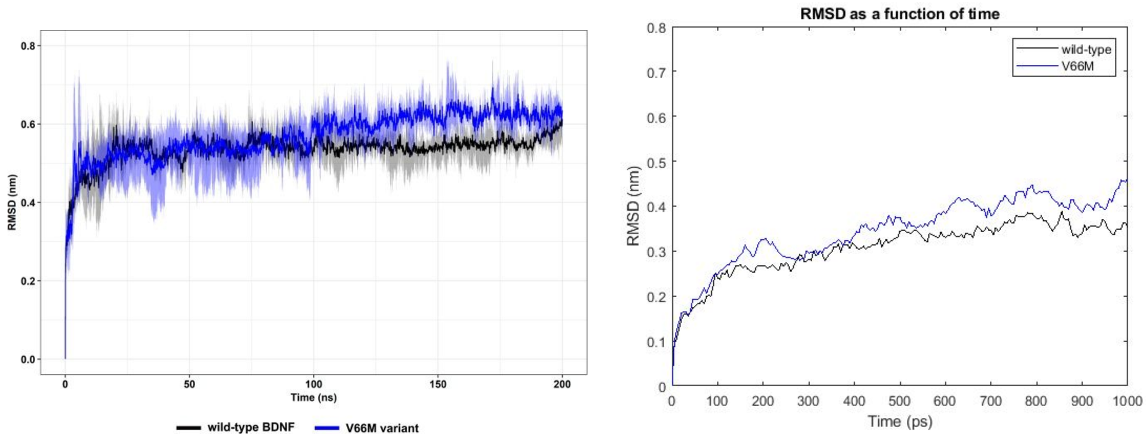


Figure3 *RMSD of wild-type BDNF and V66M variant as shown in (i) article, (ii) this project.*

Reasoning - This project performs simulations only for 1 ns whereas in the article, MD simulations are done for 200 ns. The shorter simulation time causes the differences in the two graphs, however, the zoomed-version of the RMSD plot is similar to figure (c) ensuring that the behaviour of the results is as expected. Also, the RMSD curves for the WT and the V66M variants are very similar and they coincide, as expected. Although the trend of RMSD in the first 1ns is very unclear in fig 3(i), we can interpret that the value of RMSD for the V66M variant is higher than that of the WT protein. This is observed in fig 3(ii) as well.

Root Mean Square Fluctuation(RMSF) Analysis -

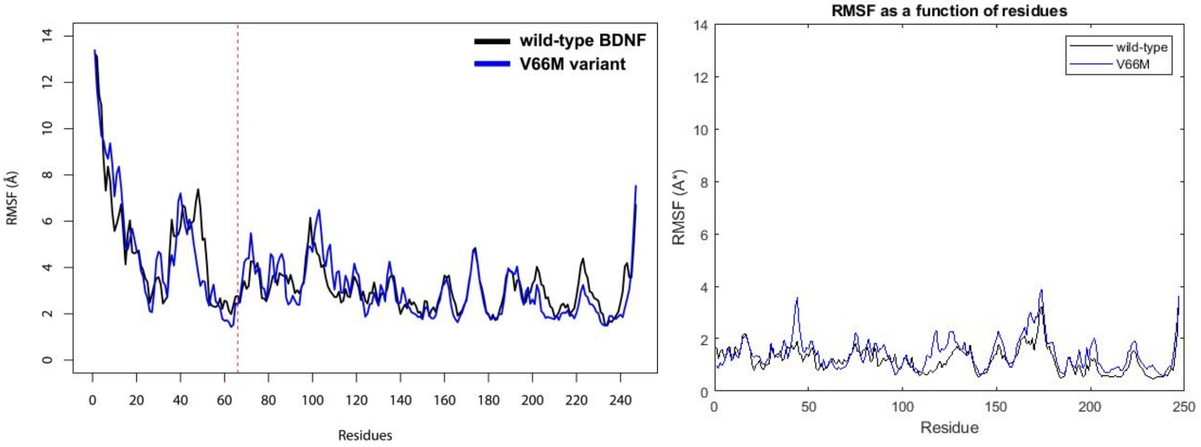


***Figure4** RMSF of wild-type BDNF and V66M variant as shown in (i) article, (ii) this project.*

Reasoning - This project performs simulations only for 1 ns whereas in the article, MD simulations are done for 200 ns. Due to the shorter simulation time we don't expect to obtain the whole of the fluctuations. Furthermore, RMSF is computed against the overall average position. This will differ for fig 4(ii) since the simulations are done only for 1 ns. Also, in figure 4(ii) the V66M fluctuates a bit more than the wild-type at ~40th residue. This could also be because the region around the 40th residue is a part of a loop (secondary structure) - Refer Supplementary Figure4. The RMSF of initial few residues also do not match and this is more likely due to shorter simulation time, which does not cover all the fluctuations as displayed in the article.

Radius of Gyration (Rg) Analysis -

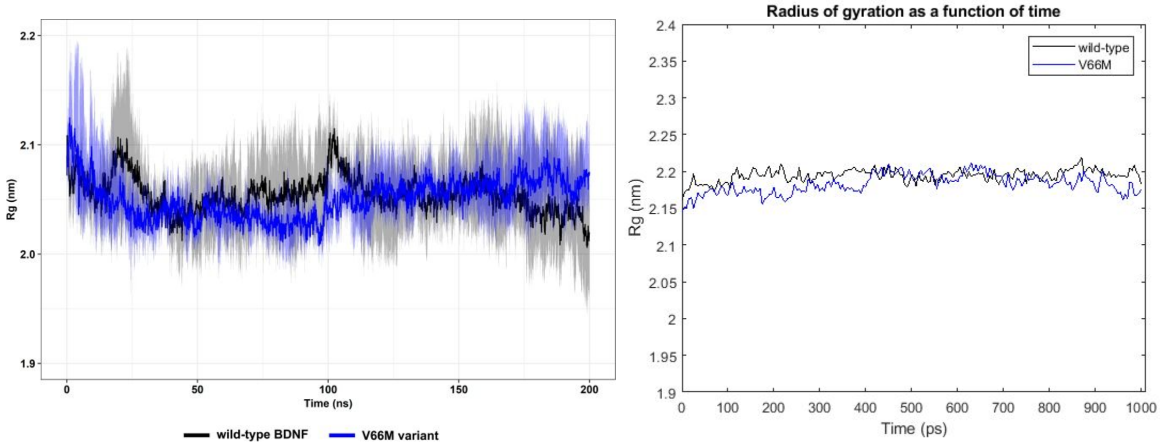


***Figure5** Rg of wild-type BDNF and V66M variant as shown in (i) article, (ii) this project.*

Reasoning - This project performs simulations only for 1 ns whereas in the article, MD simulations are done for 200 ns. If we zoom in and compare the trends of Rg for the first nanosecond, we can observe that in fig 5(ii), the values are at a higher range than in fig 5(i). This can be due to the minute differences present in both the structures since the proteins' structure files are not exactly the same as what was used in the article. Further, the article reports value and deviation for triplicates. Hence, the behavior is generalised, however, we see that the values in fig 5(ii) are still covered within the deviations in fig5(i).

Solvent Accessible Surface Area (SASA) Analysis

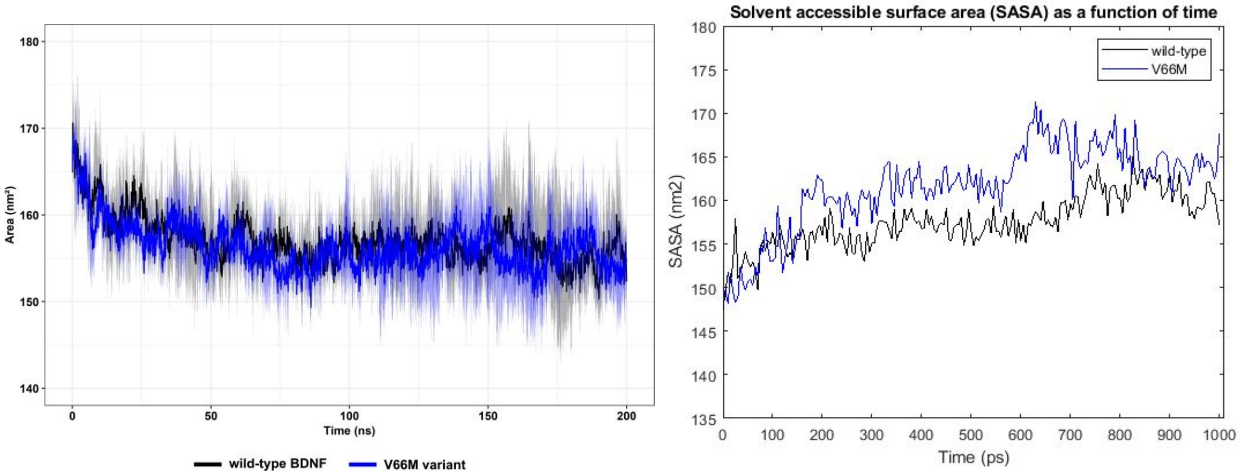


Figure6 *SASA of wild-type BDNF and V66M variant as shown in (i) article, (ii) this project.*

Reasoning - This project performs simulations only for 1 ns whereas in the article, MD simulations are done for 200 ns. Although the trend of SASA in the first 1ns is very unclear in fig 6(i), we can interpret that the value of SASA for the V66M variant is higher than that of the WT protein. This is observed in fig 6(ii) as well. Secondly, the decreasing trend in fig 6(i) indicates that the protein is vaguely moving towards a closed state from an open state, but that is across 200 ns. In fig 6(ii) we can see that the decreasing trend starts at ~700 ps. Whether the protein in this project is indeed moving towards a closed state can only be confirmed by running simulations for more than 1 ns.

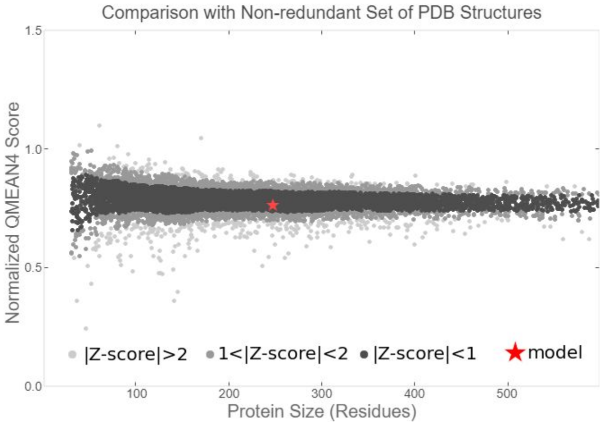
IX ACKNOWLEDGEMENTS

I thank the course faculty Dr. Hamsa Priya Mohanasundaram, for her valuable inputs throughout the course and for making this a very fun learning experience. I thank my classmates for all the meaningful discussions.

X APPENDIX

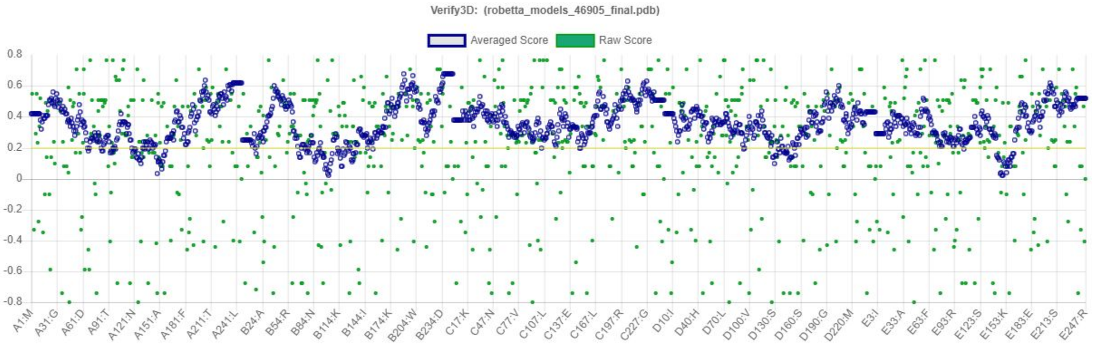
A Validation of BDNF structure

(1) QMEAN estimates the quality of the submitted model based on its physicochemical properties and then generates a value referring to the overall quality of the structure. This value is then compared against calculated QMEAN-scores of 9766 high-resolution experimental structures. For the BDNF.pdb model, the QMEAN-score was -1.13, which is comparable to that of high-resolution experimental structures.



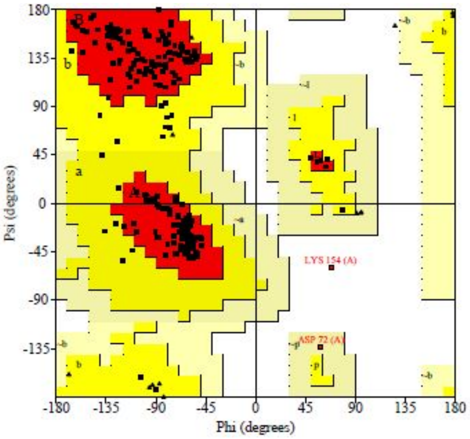
Supplementary Figure1 *Structure validation by QMEAN showing the QMEAN-score of the predicted model of BDNF (red), when compared to a non-redundant set of high-resolution experimental structures (gray and black).*

(2) Verify3D assesses the local quality of the submitted model based on its structure-sequence compatibility to generate a compatibility value for each residue of the protein.



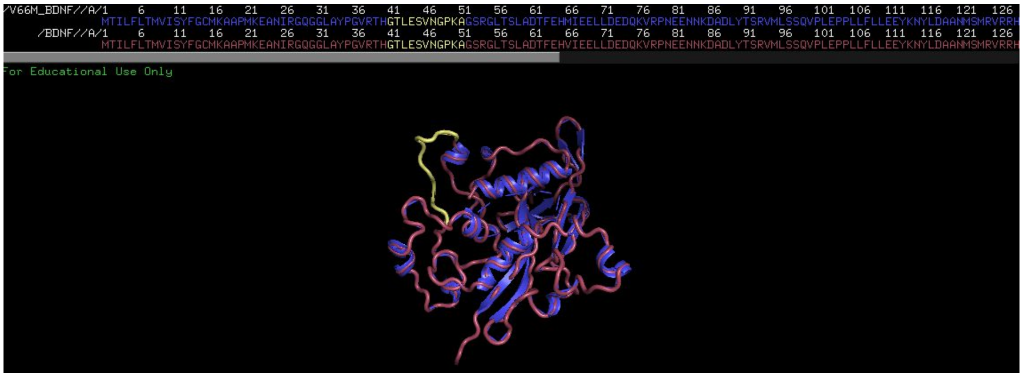
Supplementary Figure2 *Structure validation by Verify3D shows the 3D-1D score for each atom in the PDB file.*

(3) PROCHECK evaluates the stereochemical quality of the submitted models based on its phi/psi angle arrangements and then generates Ramachandran plots in which the protein residues are placed on favored, allowed and not allowed regions.



Supplementary Figure3 *PROCHECK's Ramachandran plot shows the in silico structure of BDNF. Most residues (89.4%) are located in the most favored regions, while 9.7% are in additional allowed regions, and only 0.5% in disallowed regions, configuring a high-quality model.*

B. Loop region on BDNF_WT and BDNF_V66M



```
/V66M_BDNF//A/1     6     11    16    21    26    31    36    41    46    51    56    61    66    71    76    81    86    91    96    101  106  111  116  121  126
                MTILFLTMVISYFGCMKAAPMKEANIRGQGGLAYPGVRTHGTLESVNGPKAGSRGLTSLADTFEHMIEELLDEDQKVRPNEENNKDADLYTSRVMLSSQVPLEPPLLFLLEEYKNYLDAANMSMRVRRH
     /BDNF//A/1     6     11    16    21    26    31    36    41    46    51    56    61    66    71    76    81    86    91    96    101  106  111  116  121  126
                MTILFLTMVISYFGCMKAAPMKEANIRGQGGLAYPGVRTHGTLESVNGPKAGSRGLTSLADTFEHVIEELLDEDQKVRPNEENNKDADLYTSRVMLSSQVPLEPPLLFLLEEYKNYLDAANMSMRVRRH
For Educational Use Only
```

Supplementary Figure4 *The secondary structure of the WT and V66M variants of BDNF across residues 40 - 50, correspond to loop regions. This means that they have higher flexibility in their conformation and are prone to higher fluctuations, as observed in the RMSF analysis - Fig 4(ii).*

C List of Commands

The following is the list of commands that were used in setting up the system, minimising it, running NVT, NPT and MD production simulations before going to the analysis part.

Creating the .GRO file.

- `gmx pdb2gmx -f BDNF.pdb -o BDNF.gro -ignh`

Create a PBC box and solvate

- `gmx editconf -f BDNF.gro -o BDNF_box.gro -c -d 1.0 -bt dodecahedron -box 5.50 8.27 4.56`
- `gmx solvate -cp BDNF_box.gro -cs spc216.gro -o solv.gro -p topol.top -box 5.50 8.27 4.56`

Neutralize the system

- `gmx grompp -f ions.mdp -c solv.gro -p topol.top -o ions.tpr`
- `gmx genion -s ions.tpr -o ions.gro -p topol.top -nname CL -pname NA -neutral -conc 0.15`

Minimize the protein system

- `gmx grompp -f minim.mdp -c ions.gro -p topol.top -o em.tpr`
- `gmx mdrun -v -deffnm em &`

NVT simulation

- `gmx grompp -f nvt.mdp -c em.gro -r em.gro -p topol.top -o nvt.tpr`
- `gmx mdrun -nt 8 -v -deffnm nvt`

NPT simulation - After every 2 steps, change the value of force constant in the posre.itp file.

- `gmx grompp -f npt1000.mdp -c nvt.gro -r nvt.gro -t nvt.cpt -p topol.top -o npt1000.tpr`
- `gmx mdrun -nt 8 -v -deffnm npt1000`
- `gmx grompp -f npt500.mdp -c npt1000.gro -r npt1000.gro -t npt1000.cpt -p topol.top -o npt500.tpr`
- `gmx mdrun -nt 8 -v -deffnm npt500`
- `gmx grompp -f npt250.mdp -c npt500.gro -r npt500.gro -t npt500.cpt -p topol.top -o npt250.tpr`
- `gmx mdrun -nt 8 -v -deffnm npt250`
- `gmx grompp -f npt0.mdp -c npt250.gro -r npt250.gro -t npt250.cpt -p topol.top -o npt0.tpr`
- `gmx mdrun -nt 8 -v -deffnm npt0`

MD production simulation

- `gmx grompp -f md.mdp -o md.tpr -c npt0.gro -r npt0.gro -p topol.top -t npt0.cpt`
- `gmx mdrun -v -deffnm md &`

Following minimization, NVT, NPT and MD simulations, gmx energy command can be run to further analyse the variation of several parameters throughout the simulation.

- `gmx energy -f *.edr -o **.xvg - Choose necessary parameters and press Enter`

Removing the PBC box

- `gmx trjconv -f md.xtc -o md_nopbc.xtc -s md.tpr -pbc mol -ur compact`

MD Function Analysis

- `RG - gmx gyrate -f md_nopbc.xtc -s md.tpr -o gyrate.xvg (choose 1(protein) for Rg calculations)`
- `RMSD - gmx rms -f md_nopbc.xtc -s md.tpr -o rmsd.xvg -dist rmsd_dist.xvg (Choose 4(backbone) for least square fit and 1(protein) for rmsd calculations)`
- `RMSF - gmx rmsf -f md_nopbc.xtc -s md.tpr -o rmsf.xvg -od rmsdev.xvg (choose 3(Calpha) for RMSF calculations)`
- `SASA - gmx sasa -f md_nopbc.xtc -s md.tpr -o area.xvg (Choose 1(protein) for SASA calculations)`

Note - All the important files are added and stored here.