REFERENCES

[1] Sumathi S, Sivanandam S N (2006), "Introduction *to Data Mining and its Applications*", Studies in Computational Intelligence, Vol.29, pp 1-20.

[2] Zaki J, Wang T L, Toivonen T T (2001), "*BIOKDD01: Workshop on Data Mining in Bioinformatics*", Vol.1 (2), pp.114 -118.

[3] Khalid Raza (2010), "*Application of Data mining in Bioinformatics*", Indian Journal of Computer Science and Engineering, Vol.1 (2), pp.114 -118.

[4] Flanigan K M, Dunn D (2012), "*Mutational Spectrum of DMD Mutations in Dystrophinopathy Patients: Application of Modern Diagnostic Techniques to a Large Cohort*", PMC, Vol.13 (12).

[5] Flanigan K M, Dunn D (2013), "*Nonsense mutation-associated Becker muscular dystrophy: interplay between exon definition and splicing regulatory elements within the DMD gene*", PMC.

[6] Krahn M, Bernard R (2006), "*Screening of the CAPN3 gene in patients with possible LGMD2A*", Clinical Genetics, Vol.69 (5), pp.444–449.

[7] Nevo Y, Ahituv S (2001), "*Novel mutations in the emerin gene in Israeli families*", Human Mutation, Vol.17 (6), pp. 522.

[8] Preeti Kale and Jagannath V Aghav (2014), "*Computational Methods to Infer Human Diseases",* In the Proceedings of International Work- Conference on Bioinformatics and Biomedical Engineering, (IWBBIO'14), pp. 1-14.

[9] Goh K I, Choi IG (2012) "*Exploring the human diseasome: the human disease network*", Briefings in functional genomics, Vol.11(6), pp.533–542.

[10] Fajkusova L (2001) "*Novel dystrophin mutations revealed by analysis of dystrophin mRNA:alternative splicing suppresses the phenotypic effect of a nonsense mutation*" Neuromuscular Disorders, Vol.11(2), pp.133-138.

[11] Adie E A (2005) "*Speeding disease gene discovery by sequence based candidate prioritization"*, BMC Bioinformatics, Vol.6, pp.55.

[12] Jerry Lewis (2000), "*Facts About Rare Muscular Dystrophies Congenital (CMD), Distal (DD), Emery-Dreifuss (EDMD) & Oculopharyngeal Muscular Dystrophies (OPMD)*", Muscular dystrophy foundation. Australia.

[13] Emery A E H (2002), "*The Muscular Dystrophies",* THE LANCET, Vol.359(9307), pp. 687–695.

[14] Bushby K (2009), "*Diagnosis and management of Duchenne muscular dystrophy, part 1: diagnosis, and pharmacological and psychosocial management Charcot-Marie-Tooth Disease*" U.S. DEPARTMENT OF HEALTH AND HUMAN SERVICES Public Health Service National Institutes of Health

[15] Sakthivel Murugan S.M (2013), "*Carrier detection in Duchenne muscular dystrophy using molecular methods*", Indian J Med Res, Vol.137, pp.1102-1110

[16] Agnes Jani Acsadi (2008), "*Charcot-Marie-Tooth Neuropathies: Diagnosis and Management*", Thieme Medical Publisher, Vol.28 (2).

[17] Felix F (2013), "*Effective Classification and Gene Expression Profiling for the FacioscapulohumeralMuscular Dystrophy*", PLoS ONE, Vol. 8(12), pp.e82071.

[18] Turner C, Jones D H (2008), "*The myotonic dystrophies: diagnosis and management*", Journal of Neurology, Neurosurgery and psychiatry, Vol.81(4), pp. 358-367.

[19] Baioni M T C, Ambiel C R (2010), "*Spinal muscular atrophy: diagnosis, treatment and future prospects*", Jornal de Pediatria, Vol.86 (4), pp. 261-270.

[20] Damico A (2011), "*Spinal muscular atrophy Orphanet Journal of Rare Diseases*", Orphanet Journal of Rare Diseases, Vol.6, pp.71.

[21] Anderson N L, Anderson G N (2002), "*The Human Plasma Proteome History, Character, and Diagnostic Prospects*", Molecular & Cellular Proteomics, Vol.1, pp. 845-867.

[22] Mariko Okubo (2016), "*Genetic diagnosis of Duchenne/Becker muscular dystrophy using next-generation sequencing: validation analysis of DMD mutations*", Vol.61 (6), pp.483–489.

[23] Love DR (2004), "Limb girdle muscular dystrophy: use of dHPLC and direct sequencing to detect sarcoglycan gene mutations in a New Zealand cohort",Vol.65 (1), pp.55-60.

[24] Aartsma-Rus A (2016), "*The importance of genetic diagnosis for Duchenne muscular dystrophy*", J Med Genet, Vol.53 (3), pp.145-151.

[25] Sbiti (2002), "*Analysis of Dystrophin Gene Deletions by Multiplex PCR in Moroccan Patients*", J Biomed Biotechnol, Vol.2(3), pp.158–160.

[26] Garibyan L, Avashia N (2014), "*Research Techniques Made Simple: Polymerase Chain Reaction (PCR)*", J Invest Dermatol, Vol.133(3), pp.e6.

[27] Prior T W, Bridgeman S J (2005), "*Experience and Strategy for the Molecular Testing of Duchenne Muscular Dystrophy*", J MolDiagn, Vol.7(3), pp.317–326.

[28] Nazareth S B (2015), "*Changing trends in carrier screening for genetic disease in the United States*", Prenat Diagn, Vol.35(10), pp.931–935.

[29] Reese M G (1997), "*Improved splice site detection in Genie*". J Comput Biology, Vol.4, pp.311–323.

[30] Yeo G (2004), "*Maximum entropy modeling of short sequence motifs with applications to RNA splicing signals*", J Comput Biol, Vol.11, pp. 377–394.

[31] Cartegni L (2003), "*ESEfinder: a web resource to identify exonic splicing enhancers*". Nucleic Acids Res. Vol.31 (13), pp. 3568–3571.

[32] Lim K.H (2012), "*Spliceman–a computational web server that predicts sequence variations in pre-mRNA splicing*". Bioinformatics.Vol.28. pp. 1031–1032.

[33] Woolfe A (2010), "*Genomic features defining exonic variants that modulate splicing*". Genome Biol, Vol.11(2).

[34] Desmet F O (2009), "*Human splicing finder: an online bioinformatics tool to predict splicing signals*". Nucleic Acids Res. pp.37-67.

[35] Pulido, Seoane (2010), "*Machine Learning Techniques for Single Nucleotide Polymorphism—Disease Classification Models in Schizophrenia*", Molecules, Vol.15, pp.4875-4889.

[36] Tuhin Srivastava (2012), "*Machine learning algorithms to classify spinal muscular atrophy subtypes*", Neurology, Vol.79, 358-364.

[37] Gonza F F (2013), "*Effective Classification and Gene Expression Profiling for the Facioscapulohumeral Muscular Dystrophy*", PLoS ONE, Vol.8 (12).

[38] Chen Wang (2012), "*Computational Analysis of Muscular Dystrophy Sub-types Using A Novel Integrative Scheme*", Neurocomputing, Vol.92, pp. 9–17.

[39] Jianmin Ma (2009), "*Gene Classification using Codon Usage and SVMs*", IEEE, Vol. 6, pp. 134-143.

[40] Nisha C M (2012), "*SVM model for classification of genotypes of HCV using Relative Synonymous Codon Usage*", Journal of Advanced Bioinformatics Applications and Research, Vol 3(3), pp.357-363.

[41] Zou Q (2013), "*An approach for identifying cytokines based on a novel Ensemble classifier"*, BioMed Research International, Vol.2013 (2013).

[42] Kalari KR (2006), "*Computational approach to identify deletions or duplications within a gene*". PhD (Doctor of Philosophy) thesis, University of Iowa.

[43] Wu J (2013), "*Comparative Study of Ensemble Learning Approaches in the Identification of Disease Mutations*". 3rd International conference on Biomedical Enginering and Informatics (BMEI 2010), pp. 2306 – 2310.

[44] Mort M (2014), "*MutPred Splice: machine learning-based prediction of exonic variants that disrupt splicing*". Genome Biology, pp.15-19.

[45] Dandan Mo (2012), "*A survey on deep learning: one small step toward AI*".

[46] Chen X W (2014), "*Big Data Deep Learning: Challenges and Perspectives*" , IEEE. Translations and content mining, IEEE, Vol.2, pp.514 -525.

[47] Dean J (2012), "*Large scale distributed deep networks*," in Proc. Adv. NIPS, pp. 1232-1240.

[48] Coats (2013), "*Deep Learning with COTS HPS systems*," J. Mach. Learn. Res., Vol. 28 (3), pp.1337-1345.

[49] Huma Lodhi (2012), "*Computational biology perspective: kernel methods and deep learning*", WIREsComput Stat, Vol.4, pp.455–465.

[50] Liu (2014), "*Early diagnosis of Alzheimer's disease with deep learning*", Biomedical Imaging (ISBI), 2014 IEEE 11th International Symposium on. IEEE, pp.1015–1018.

[51] Devinder Kumar (2015), "*Lung Nodule Classification Using Deep Features in CT Images*".

[52] Marios Anthimopoulos (2016), " *Lung Pattern Classification for Interstitial Lung Diseases Using a Deep Convolutional Neural Network",* IEEE Trans Med Imaging, Vol. 35 (5), pp.1207-1216.

[53] Haofu Liao, "*A Deep Learning Approach to Universal Skin Disease Classification*", CSC 400 - Graduate Problem Seminar- project report.

[54] Yoshua Bengio (2009), "*Learning Deep Architectures for AI*", Foundations and Trends in Machine Learning, Vol. 2(1), pp.1–127.

[55] Aliper A (2016), "*Deep Learning Applications for Predicting Pharmacological Properties of Drugs and Drug Repurposing Using Transcriptomic Data*", Mol. Pharmaceutics, Vol.13 (7), pp. 2524-2530.

[56] Rashmi Tripathi (2016), "*DeepLNC, a long non-coding RNA prediction tool using deep neural network*", Netw Model Anal Health Inform Bioinforma, Vol.5 (21).

[57] Zhao Z (2016), "*A protein–protein interaction extraction approach based on deep neural network*", Int. J. Data Mining and Bioinformatics, Vol. 15 (2).

[58] Daniel Quang (2015) "*DANN: a deep learning approach for annotating the pathogenicity of genetic variants*", Bioinformatics Advance Access, Vol.31(5), pp. 761–763.

[59] Lanchantin J (2016), "*Deep GDashboard: Visualizing and Understanding Genomic Sequences Using Deep Neural Networks*".

[60] Liu, You (2015), "*Cough event classification by pretrained deep neural network*", BMC Medical informatics and decision making, Vol.15.

[61] Mohammed Abo-Zahhad (2014), "*Integrated Model of DNA Sequence Numerical Representation and Artificial Neural Network for Human Donor and Acceptor Sites Prediction*",I.J. Information Technology and Computer Science, Vol.08, pp.51-57.

[62] Nguyen N G (2016), "*DNA Sequence Classification by Convolutional Neural Network*", J. Biomedical Science and Engineering, Vol.9, pp.280-286.

[63] Abo-Zahhad M (2013), "*A New Numerical Mapping Technique for Recognition of Exons and Introns in DNA Sequences*". The 30th National Radio Science Conference, NTI, Cairo, Egypt, pp.573-580

[64] Alex Smola and Vishwanathan S V N (2008), "*Introduction to Machine Learning*", Cambridge University Press, pp.234.

[65] Nilsson N J (1998), "*Introduction to Machine Learning*", Stanford University.

[66] Witten E H, Frank E, Hall M A (2011), "*Data mining – Practical machine learning tools and techniques*", Morgan Kaufmann Publishers, Elsevier.

[67] Tom M. Mitchell (1997), "*Machine Learning*", McCraw-Hill, Boston.

[68] Christoper M. Bishop (2008), "*Pattern Recognition and Machine Learning*". Springer-verlag.

[69] Gunnar Raetsch (2003), "*A Brief introduction into Machine Learning*", Technical Report Fredrich Miescher Laboratory, Germany.

[70] Ha J, Kamber M (2006), "*Data Mining Concepts and Techniques*", Second Edition. Morgan Kaufmann publishers, San Francisco.

[71] Aruna devi R, Nirmala K (2013), "*Construction of Decision Tree : Attribute Selection Measures*", International Journal of Advancements in Research & Technology, Vol.2(4), pp.343-346.

[72] (http://www.saedsayad.com/naive_bayesian.htm)

[73] Soman K P, Loganathan R, Ajay V (2009), "*Machine Learning with SVM and other Kernal methods*". PHI, India.

[74] Benhard S, Smola A J (2002), "*Learning with Kernels: Support Vector machines, Regularization, Optimization and Beyond*". MIT press, Cambridge, MA, USA.

[75] Dietterich T G (2000), "*Ensemble Methods in Machine Learning*", Oregon State University.

[76] Gulisong (2010), "*A Triple-Random Ensemble Classification Method for Mining Multi-label Data*", 2010 IEEE International Conference on Data Mining Workshops.

[77] Cheng L (2013), "*Sampled-databased average consensus of second-order integral multi-agent systems: switching topologies and communication noises*", Automatica, Vol.49 (5), pp.1458–1464.

[78] Chen (2014), "*LibD3C: Ensemble classifiers with a clustering and a dynamic strategy*", Elseiver's Neurocomputing 123, pp. 424-435.

[79] Hao H W (2011),"*Dynamics election and circulating combination for multiple classifier systems*", Acta Automatica Sinica, Vol.37 (11), pp.1290–1295.

[80] Hao H (2003), "*Comparison of genetic algorithm and sequential search methods for classifier subset selection*", In: Proceedings of the Seventh International Conference on Document Analysis and Recognition, Cite seer, pp.765.

[81] Yan R (2007), "*Model-shared subspace boosting for multi-label classification*", In: Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM, pp .834–843.

[82] Zhou Z H (2002), "*Ensembling neural networks: many could be better than all*," Artificial Intelligence, Vol. 137(1-2), pp. 239–263.

[83] Marina Sokolova a, Guy Lapalme (2009), "*A systematic analysis of performance measures for classification tasks Information Processing and Management*", Vol.45 (4), pp 427–437.

[84] Li Deng (2017), "*An Overview of Deep-Structured Learning for Information Processing*" Microsoft Research, Redmond, WA 98052, USA.

[85] Chen X W, Lin X (2014), "*Big Data Deep Learning: Challenges and Perspectives*", IEEE Access, Vol. 2, pp.514 – 525.

[86] Bengio Y (2009), "*Learning Deep Architectures*", ICML Workshop on Learning Feature Hierarchies, Montreal.

[87] Castrounis A (2016), "*Artificial Intelligence, Deep Learning, and Neural Networks Explained* ", Inno Archi Tech newsletter

[88] Bengio (2009), "*Learning Deep Architectures for AI*", Foundations and Trends in Machine Learning, Vol. 2(1), pp.1–127.

[89] Schmidhuber J (2015), "*Deep Learning in Neural Networks: An Overview*", Neural Networks,Vol.61, pp.85–117

[90] Szegedy, Christian, Alexander Toshev, Dumitru Erhan (2013), "*Deep neural networks for object detection*", Advances in Neural Information Processing Systems.

[91] Hochreiter, Sepp (1997), "*Long Short-Term Memory, Neural Computation*", Vol.9 (8), pp. 1735–1780

[92] Gers, Felix, Schraudolph (2002), "*Learning precise timing with LSTM recurrent networks*", Journal of Machine Learning Research, Vol.3, pp.115–143.

[93] Dan Gillick, Cliff Brunk (2015), "*Multilingual Language Processing From Bytes*".

[94] Wen T S (2013), "*Recurrent neural network based language model Personalization by Social Network Crowdsourcing*," Interspeech, pp. 2703-2707.

[95] LeCun (1998), "*Gradient-based learning applied to document recognition*". Proceedings of the IEEE, Vol. 86 (11), pp.2278–2324.

[96] T. Sainath (2013), "*Convolutional neural networks for LVCSR*", ICASSP.

[97] Maryam M N (2015), "*Deep learning applications and challenges in big data analytics*", Journal of Big Data, Springer, Vol.2(1).

[98] P Singhal, "*Sentiment Analysis and Deep Learning: A Survey*"

[99] Bengio Y (2009), "*Learning Deep Architectures for AI*", Foundations and Trends in Machine Learning, Vol. 2 (1), pp.1–127.

[100] (http://ufldl.stanford.edu/tutorial/supervised/OptimizationStochasticGradientDescent/)

[101] (http://ruder.io/optimizing-gradient-descent/)

[102] Panchal G (2011), "*Behaviour Analysis of Multilayer Perceptrons with Multiple Hidden Neurons and Hidden Layers*", International Journal of Computer Theory and Engineering, Vol. 3 (2).

[103] Hochreiter S (2001), "*Gradient flow in recurrent nets: the difficulty of learning long-term dependencies*". A Field Guide to Dynamical Recurrent Networks. John Wiley & Sons.

[104] Bengio Y (2013), "*Advances in optimizing recurrent networks*", 2013 IEEE International Conference on Acoustics, Speech and Signal Processing, pp. 8624–8628.

[105] Hinton G. E (2012). "*Improving neural networks by preventing co-adaptation of feature detectors*"

[106] Liou (2008), "*Modeling word perception using the Elman network*", Neurocomputing, Vol.71, pp.3150–3157.

[107] Liou (2014), "*Autoencoder for Words*", Neurocomputing, Vol.139, pp.84–96.

[108] Kingma D P (2013), "*Auto-Encoding Variational Bayes*".

[109] Boesen A (2015), "*Generating Faces with Torch*".

[110] Rumelhart D E (1986), "*Learning internal representations by error propagation*", In Parallel Distributed Processing, Vol.1, Foundations, MIT Press,Vol.1, pp.318-362.

[111] Hebb D O (1949), "*The organization of behavior: A neuropsychological study*", Wiley Inter science, New York, Vol.11 (7-8), pp 1531-1549.

[112] Oja E (1982), "*Simplified neuron model as a principal component analyzer. Journal of mathematical biology*", Vol.15 (3), pp.267–273.

[113] Hinton, (2006). "*A fast learning algorithm for deep belief nets. Neural Computation*", Vol.18 (7), pp.1527–1554.

[114] Hinton GE (1994), "*Autoencoders, minimum description length, and Helmholtz free energy*", Adv. Neural Inf.Process.Syst, Vol.6, pp. 3-10.

[115] Cowan J D (1994), "*Advances in Neural Information Processing Systems 5*", Morgan Kaufmann Publishers, pp. 836-844.

[116] Q. Wang (1996), "*Positional cloning of a novel potassium channel gene: KVLQT1 mutations cause cardiac arrhythmias*", Nature Genetics, Vol.12, pp.17 – 23.

[117] Li-li Pan (2015), "*Positional cloning and next-generation sequencing identified a TGM6 mutation in a large Chinese pedigree with acute myeloid leukaemia*", European Journal of Human Genetics,Vol. 23, pp.218-223

[118] Amberger J S (2015), "*OMIM.org: Online Mendelian Inheritance in Man (OMIM), an online catalog of human genes and genetic disorders*", Nucleic Acids Res, Vol.43, pp. D789–D798

[119] Stenson P D (2013), "*The Human Gene Mutation Database: building a comprehensive mutation repository for clinical and molecular genetics, diagnostic testing and personalized genomic medicine*", Human Genetics, Vol.133(1), pp.1-9.

[120] (https://www.ncbi.nlm.nih.gov/)

LIST OF PUBLICATIONS

Workshops/Seminar Attended

1. National level workshop on “Big Data Analytics”, organized by Anna University - Regional Centre, Coimbatore, from October 18-19, 2013.
2. National level workshop on “Big Data Analytics Tools” organized by PSGR Krishnammal College for women, Coimbatore, from January 24-25, 2014.
3. National workshop on “Genome Computing (GenComp 2014)” organized by department of Computer Science, Periyar university, Salem, from Janurary 29 – 30, 2014.
4. National seminar on “Big data and cloud computing for Bioinformatics Applications” organized by Kongu Engineering college, Erode, from January 9-10, 2015.
5. National level workshop on “Research Methods and Research Directions in Computer Science” organized by PSGR Krishnammal College for women, Coimbatore, from February 18-19, 2015.
6. Workshop on “Big Data Analytics”, organized by ACM student chapter, ISI Kolkata and Indian Statistical Institute, Kolkata, from August 20-21, 2015.
7. International workshop on “Computational techniques for Wetlab Data Analysis”, organized by PG & Research Department of Biotechnology, National College, Trichy, from September 22-25, 2015.
8. Workshop on “Deep Learning”, organized by Department of Computer Science and Engineering, PSG College of Technology, Coimbatore on October 7, 2016.

Papers presented in National Conferences

1. “Codon Optimization using Pattern Matching”, Machine Learning: Challenges and Opportunities Ahead”, GRG School of Applied Computer Technology, February 2014.
2. “Muscular Dystrophy Disease Prediction using Support Vector Machine”, National Conference on Intelligent Computing and Data Analytics, Department of Information Science and Technology, CEG Campus, Anna University, March 2016.

Papers published in International conference proceedings

1. “Predicting Muscular Dystrophy with Sequence based Features for Point Mutations”, IEEE Conference on research in Computational Intelligence and communication Network, IEEE CIS Kolkata chapter, Nov 2015, ISBN 978-1-4673-6734-9, pp: 235 - 240
2. “Predicting Muscular Dystrophy through Genetic testing – A Study”, International Conference on Innovative trends in Electronics Communication and Applications, ASDF and IIT Madras Research park, Chennai, Dec 2015,ISBN978-81-929742-6-2. Vol – 01, pp: 65-71
3. “Ensemble Learning for identifying Muscular dystrophy diseases using codon bias pattern”, Proceedings of the 5th International Conference on Frontiers in Intelligent Computing: Theory and Applications, Advances in Intelligent Systems and Computing 515, ISBN 978-981-10-3152-6, Vol 1, pp: 21-29, Springer (AISC) series (**Scopus Indexed**)

Papers published/accepted in International journals

1. "Muscular Dystrophy Disease Classification Using Relative Synonymous Codon Usage", International Journal of Machine Learning and Computing vol.6, no. 2,ISSN- 2010-3700, pp. 139-144, 2016 (**Published – Google Scholar Indexed**).
2. “Identification of Rare Genetic Disorder from Single Nucleotide Variants Using Supervised Learning Technique”, International journal of control theory and applications Vol.9, no.34, pp. 801-810, 2016 (**Published - Scopus indexed**).
3. "Shallow Learning model for diagnosing neuromuscular disorder from splicing variants", World Journal of Engineering, Vol. 14 Issue: 4, pp.329-336, 2017 (**Published -Scopus Indexed, ISI indexed**).
4. "Prognosis of Muscular dystrophy disorder with Extrinsic and intrinsic descriptors through ensemble learning", Turkish journal of Electrical Engineering and computer sciences (**Accepted - SCIE Indexed**).
5. "Data Driven Approach for Genetic Disorder Prediction by Aggregating Mutational Features", Asian Journal of Information Technology (**Accepted**).
6. "Nucleotide and codon mapping schemes for deep learning to diagnose muscular dystrophy", Frontiers in Biosciences (**Accepted - SCIE Indexed**)

Papers in review - International journals

1. "Identification of Muscular Dystrophy with Mutation Based Features through Shallow and Deep Learning". Journal of Biomedical Informatics – Elsevier publications.

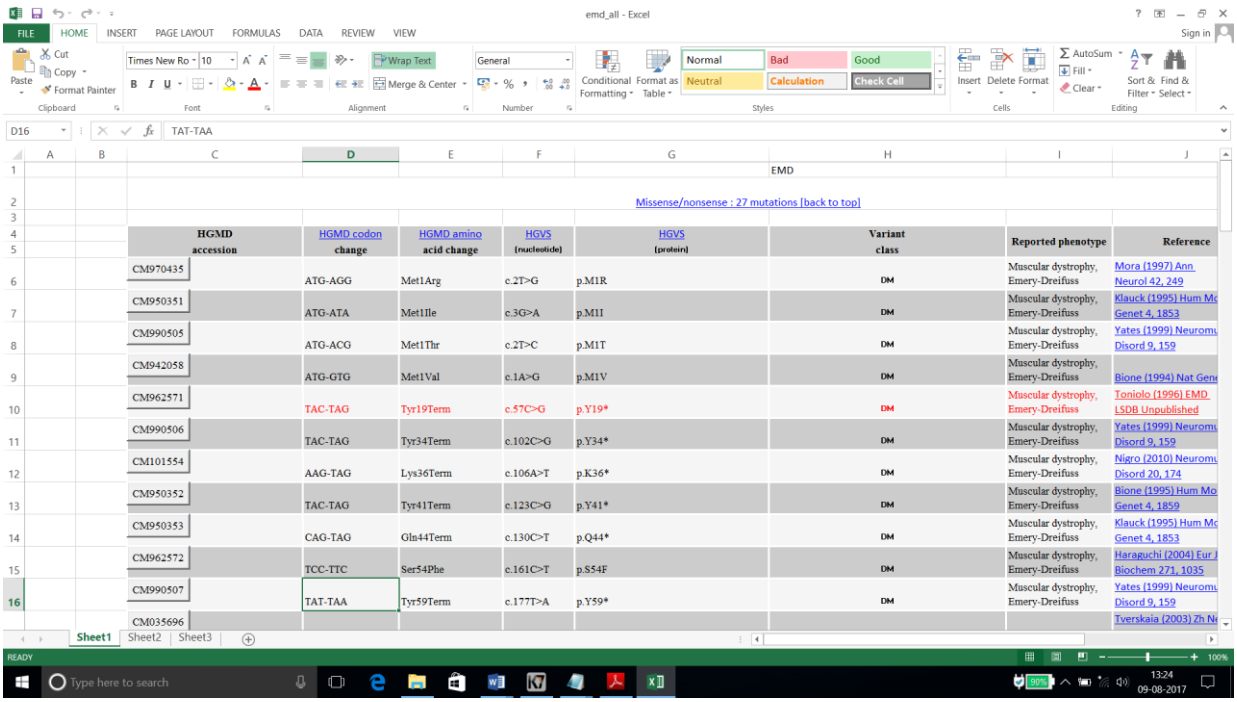
Appendix – A

Sample Gene Sequences

cDNA sequence for EMD gene

>gi|195234784|ref|NM_000117.2| Homo sapiens emerin (EMD), mRNA
ATGGACAACTACGCAGATCTTTCGGATACCGAGCTGACCACCTTGCTGCGCCGGTACAAC
ATCCCGCACGGGCCTGTAGTAGGATCAACTCGTAGGCTTTACGAGAAGAAGATCTTCGAG
TACGAGACCCAGAGGCGGCGGCTCTCGCCCCCCAGCTCGTCCGCCGCCTCCTCTTATAGC
TTCTCTGACTTGAATTCGACTAGAGGGGATGCAGATATGTATGATCTTCCCAAGAAAGAG
GACGCTTTACTCTACCAGAGCAAGGGCTACAATGACGACTACTATGAAGAGAGCTACTTC
ACCACCAGGACTTATGGGGAGCCCGAGTCTGCCGGCCCGTCCAGGGCTGTCCGCCAGTCA
GTGACTTCATTCCCAGATGCTGACGCTTTCCATCACCAGGTGCATGATGACGATCTTTTG
TCTTCTTCTGAAGAGGAGTGCAAGGATAGGGAACGCCCCATGTACGGCCGGGACAGTGCC
TACCAGAGCATCACGCACTACCGCCCTGTTTCAGCCTCCAGGAGCTCCCTGGACCTGTCC
TATTATCCTACTTCCTCCTCCACCTCTTTTATGTCCTCCTCATCATCTTCCTCTTCATGG
CTCACCCGCCGTGCCATCCGGCCTGAAAACCGTGCTCCTGGGGCTGGGCTGGGCCAGGAT
CGCCAGGTCCCGCTCTGGGGCCAGCTGCTGCTTTTCCTGGTCTTTGTGATCGTCCTCTTC
TTCATTTACCACTTCATGCAGGCTGAAGAAGGCAACCCCTTCTAG

Missense mutation information for EMD gene in HGMD



emd_all - Excel

EMD

Missense/nonsense : 27 mutations [back to top]

HGMD accession	HGMD codon change	HGMD amino acid change	HGVS (nucleotide)	HGVS (protein)	Variant class	Reported phenotype	Reference
CM970435	ATG-AGG	Met1Arg	c.2T>G	p.M1R	DM	Muscular dystrophy, Emery-Dreifuss	Mora (1997) Ann Neurol 42, 249
CM950351	ATG-ATA	Met1Ile	c.3G>A	p.M1I	DM	Muscular dystrophy, Emery-Dreifuss	Klauck (1995) Hum Mo Genet 4, 1853
CM990505	ATG-ACG	Met1Thr	c.2T>C	p.M1T	DM	Muscular dystrophy, Emery-Dreifuss	Yates (1999) Neurom Disord 9, 159
CM942058	ATG-GTG	Met1Val	c.1A>G	p.M1V	DM	Muscular dystrophy, Emery-Dreifuss	Bione (1994) Nat Gen
CM962571	TAC-TAG	Tyr19Term	c.57C>G	p.Y19*	DM	Muscular dystrophy, Emery-Dreifuss	Toniolo (1996) EMD LSDB Unpublished
CM990506	TAC-TAG	Tyr34Term	c.102C>G	p.Y34*	DM	Muscular dystrophy, Emery-Dreifuss	Yates (1999) Neurom Disord 9, 159
CM101554	AAG-TAG	Lys36Term	c.106A>T	p.K36*	DM	Muscular dystrophy, Emery-Dreifuss	Nigro (2010) Neurom Disord 20, 174
CM950352	TAC-TAG	Tyr41Term	c.123C>G	p.Y41*	DM	Muscular dystrophy, Emery-Dreifuss	Bione (1995) Hum Mo Genet 4, 1859
CM950353	CAG-TAG	Gln44Term	c.130C>T	p.Q44*	DM	Muscular dystrophy, Emery-Dreifuss	Klauck (1995) Hum Mo Genet 4, 1853
CM962572	TCC-TTC	Ser54Phe	c.161C>T	p.S54F	DM	Muscular dystrophy, Emery-Dreifuss	Haraguchi (2004) Eur J Biochem 271, 1035
CM990507	TAT-TAA	Tyr59Term	c.177T>A	p.Y59*	DM	Muscular dystrophy, Emery-Dreifuss	Yates (1999) Neurom Disord 9, 159
CM035696							Tverskaia (2003) Zh Ne

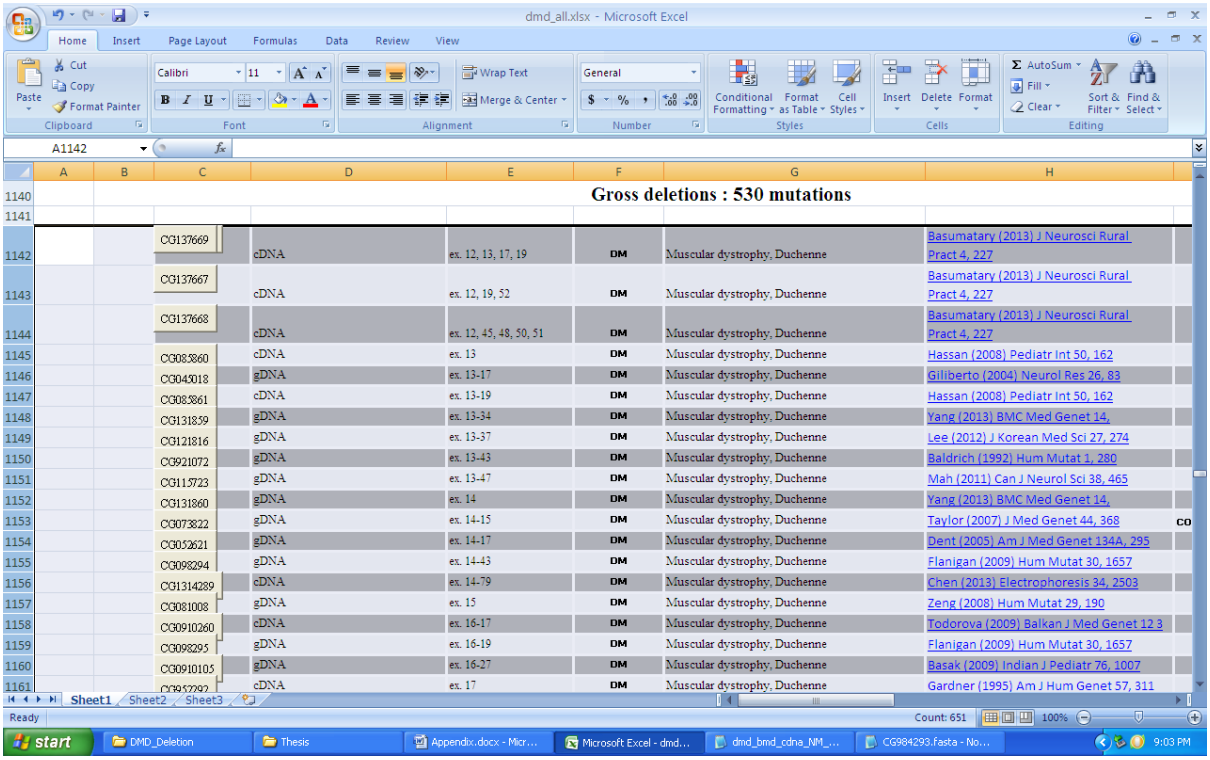
Missense mutated gene sequence for EMD – CM970435

>gi|195234784|ref|NM_000117.2| Homo sapiens emerin (EMD), mRNA
AGGGACAACTACGCAGATCTTTCGGATACCGAGCTGACCACCTTGCTGCGCCGGTACAAC
ATCCCGCACGGGCCTGTAGTAGGATCAACTCGTAGGCTTTACGAGAAGAAGATCTTCGAG
TACGAGACCCAGAGGCGGCGGCTCTCGCCCCCCAGCTCGTCCGCCGCCTCCTCTTATAGC
TTCTCTGACTTGAATTCGACTAGAGGGGATGCAGATATGTATGATCTTCCCAAGAAAGAG
GACGCTTTACTCTACCAGAGCAAGGGCTACAATGACGACTACTATGAAGAGAGCTACTTC
ACCACCAGGACTTATGGGGAGCCCGAGTCTGCCGGCCCGTCCAGGGCTGTCCGCCAGTCA
GTGACTTCATTCCCAGATGCTGACGCTTTCCATCACCAGGTGCATGATGACGATCTTTTG
TCTTCTTCTGAAGAGGAGTGCAAGGATAGGGAACGCCCCATGTACGGCCGGGACAGTGCC
TACCAGAGCATCACGCACTACCGCCCTGTTTCAGCCTCCAGGAGCTCCCTGGACCTGTCC
TATTATCCTACTTCCTCCTCCACCTCTTTTATGTCCTCCTCATCATCTTCCTCTTCATGG
CTCACCCGCCGTGCCATCCGGCCTGAAAACCGTGCTCCTGGGGCTGGGCTGGGCCAGGAT
CGCCAGGTCCCGCTCTGGGGCCAGCTGCTGCTTTTCCTGGTCTTTGTGATCGTCCTCTTC
TTCATTTACCACTTCATGCAGGCTGAAGAAGGCAACCCCTTCTAG

cDNA sequence for DMD gene

>gi|238018044|ref|NM_004006.2| Homo sapiens dystrophin (DMD), transcript variant Dp427m, mRNA
ATGCTTTGGTGGGAAGAAGTAGAGGACTGTTATGAAAGAGAAGATGTTCAAAAGAAAACA
TTCACAAAATGGGTAAATGCACAATTTTCTAAGTTTGGGAAGCAGCATATTGAGAACCTC
TTCAGTGACCTACAGGATGGGAGGCGCCTCCTAGACCTCCTCGAAGGCCTGACAGGGCAA
AAACTGCCAAAAGAAAAAGGATCCACAAGAGTTCATGCCCTGAACAATGTCAACAAGGCA
CTGCGGGTTTTGCAGAACAATAATGTTGATTTAGTGAATATTGGAAGTACTGACATCGTA
GATGGAAATCATAAACTGACTCTTGGTTTGATTTGGAATATAATCCTCCACTGGCAGGTC
AAAAATGTAATGAAAAATATCATGGCTGGATTGCAACAAACCAACAGTGAAAAGATTCTC
CTGAGCTGGGTCCGACAATCAACTCGTAATTATCCACAGGTTAATGTAATCAACTTCACC
ACCAGCTGGTCTGATGGCCTGGCTTTGAATGCTCTCATCCATAGTCATAGGCCAGACCTA
TTTGACTGGAATAGTGTGGTTTGCCAGCAGTCAGCCACACAACGACTGGAACATGCATTC
AACATCGCCAGATATCAATTAGGCATAGAGAAACTACTCGATCCTGAAGATGTTGATACC
ACCTATCCAGATAAGAAGTCCATCTTAATGTACATCACATCACTCTTCCAAGTTTTGCCT
CAACAAGTGAGCATTGAAGCCATCCAGGAAGTGGAAATGTTGCCAAGGCCACCTAAAGTG
ACTAAAGAAGAACATTTTCAGTTACATCATCAAATGCACTATTCTCAACAGATCACGGTC
AGTCTAGCACAGGGATATGAGAGAACTTCTTCCCCTAAGCCTCGATTCAAGAGCTATGCC
TACACACAGGCTGCTTATGTCACCACCTCTGACCCTACACGGAGCCCATTTCCTTCACAG
CATTTGGAAGCTCCTGAAGACAAGTCATTTGGCAGTTCATTGATGGAGAGTGAAGTAAAC
CTGGACCGTTATCAAACAGCTTTAGAAGAAGTATTATCGTGGCTTCTTTCTGCTGAGGAC
ACATTGCAAGCACAAGGAGAGATTTCTAATGATGTGGAAGTGGTGAAAGACCAGTTTCAT
ACTCATGAGGGGTACATGATGGATTTGACAGCCCATCAGGGCCGGGTTGGTAATATTCTA
CAATTGGGAAGTAAGCTGATTGGAACAGGAAAATTATCAGAAGATGAAGAAACTGAAGTA
CAAGAGCAGATGAATCTCCTAAATTCAAGATGGGAATGCCTCAGGGTAGCTAGCATGGAA
AAACAAAGCAATTTACATAGAGTTTTAATGGATCTCCAGAATCAGAAACTGAAAGAGTTG
AATGACTGGCTAACAAAAACAGAAGAAAGAACAAGGAAAATGGAGGAAGAGCCTCTTGGA
CCTGATCTTGAAGACCTAAAACGCCAAGTACAACAACATAAGGTGCTTCAAGAAGATCTA
GAACAAGAACAAGTCAGGGTCAATTCTCTCACTCACATGGTGGTGGTAGTTGATGAATCT
AGTGGAGATCACGCAACTGCTGCTTTGGAAGAACAACTTAAGGTATTGGGAGATCGATGG
GCAAACATCTGTAGATGGACAGAAGACCGCTGGGTTCTTTTACAAGACATCCTTCTCAAA
TGGCAACGTCTTACTGAAGAACAGTGCCTTTTTAGTGCATGGCTTTCAGAAAAAGAAGAT
GCAGTGAACAAGATTCACACAACTGGCTTTAAAGATCAAAATGAAATGTTATCAAGTCTT
CAAAAACTGGCCGTTTTAAAAGCGGATCTAGAAAAGAAAAAGCAATCCATGGGCAAACTG
TATTCACTCAAACAAGATCTTCTTTCAACACTGAAGAATAAGTCAGTGACCCAGAAGACG
GAAGCATGGCTGGATAACTTTGCCCGGTGTTGGGATAATTTAGTCCAAAAACTTGAAAAG
AGTACAGCACAGATTTCACAGGCTGTCACCACCACTCAGCCATCACTAACACAGACAACT
GTAATGGAAACAGTAACTACGGTGACCACAAGGGAACAGATCCTGGTAAAGCATGCTCAA
GAGGAACTTCCACCACCACCTCCCCAAAAGAAGAGGCAGATTACTGTGGATTCTGAAATT
AGGAAAAGGTTGGATGTTGATATAACTGAACTTCACAGCTGGATTACTCGCTCAGAAGCT
GTGTTGCAGAGTCCTGAATTTGCAATCTTTCGGAAGGAAGGCAACTTCTCAGACTTAAAA

Gross deletion mutation information for DMD gene in HGMD



dmd_all.xlsx - Microsoft Excel

Gross deletions : 530 mutations

CG137669	cDNA	ex. 12, 13, 17, 19	DM	Muscular dystrophy, Duchenne	Basumatary (2013) J Neurosci Rural Pract 4, 227
CG137667	cDNA	ex. 12, 19, 52	DM	Muscular dystrophy, Duchenne	Basumatary (2013) J Neurosci Rural Pract 4, 227
CG137668	cDNA	ex. 12, 45, 48, 50, 51	DM	Muscular dystrophy, Duchenne	Basumatary (2013) J Neurosci Rural Pract 4, 227
CG083860	cDNA	ex. 13	DM	Muscular dystrophy, Duchenne	Hassan (2008) Pediatr Int 50, 162
CG043018	gDNA	ex. 13-17	DM	Muscular dystrophy, Duchenne	Giliberto (2004) Neurol Res 26, 83
CG083861	cDNA	ex. 13-19	DM	Muscular dystrophy, Duchenne	Hassan (2008) Pediatr Int 50, 162
CG131859	gDNA	ex. 13-34	DM	Muscular dystrophy, Duchenne	Yang (2013) BMC Med Genet 14,
CG121816	gDNA	ex. 13-37	DM	Muscular dystrophy, Duchenne	Lee (2012) J Korean Med Sci 27, 274
CG921072	gDNA	ex. 13-43	DM	Muscular dystrophy, Duchenne	Baldrich (1992) Hum Mutat 1, 280
CG115723	gDNA	ex. 13-47	DM	Muscular dystrophy, Duchenne	Mah (2011) Can J Neurol Sci 38, 465
CG131860	gDNA	ex. 14	DM	Muscular dystrophy, Duchenne	Yang (2013) BMC Med Genet 14,
CG073822	gDNA	ex. 14-15	DM	Muscular dystrophy, Duchenne	Taylor (2007) J Med Genet 44, 368
CG052621	gDNA	ex. 14-17	DM	Muscular dystrophy, Duchenne	Dent (2005) Am J Med Genet 134A, 295
CG098294	gDNA	ex. 14-43	DM	Muscular dystrophy, Duchenne	Flanigan (2009) Hum Mutat 30, 1657
CG1314289	cDNA	ex. 14-79	DM	Muscular dystrophy, Duchenne	Chen (2013) Electrophoresis 34, 2503
CG081008	gDNA	ex. 15	DM	Muscular dystrophy, Duchenne	Zeng (2008) Hum Mutat 29, 190
CG0910260	cDNA	ex. 16-17	DM	Muscular dystrophy, Duchenne	Todorova (2009) Balkan J Med Genet 12 3
CG098295	gDNA	ex. 16-19	DM	Muscular dystrophy, Duchenne	Flanigan (2009) Hum Mutat 30, 1657
CG0910105	gDNA	ex. 16-27	DM	Muscular dystrophy, Duchenne	Basak (2009) Indian J Pediatr 76, 1007
CG952292	cDNA	ex. 17	DM	Muscular dystrophy, Duchenne	Gardner (1995) Am J Hum Genet 57, 311

Gross deletion mutated gene sequence for DMD – CG137668

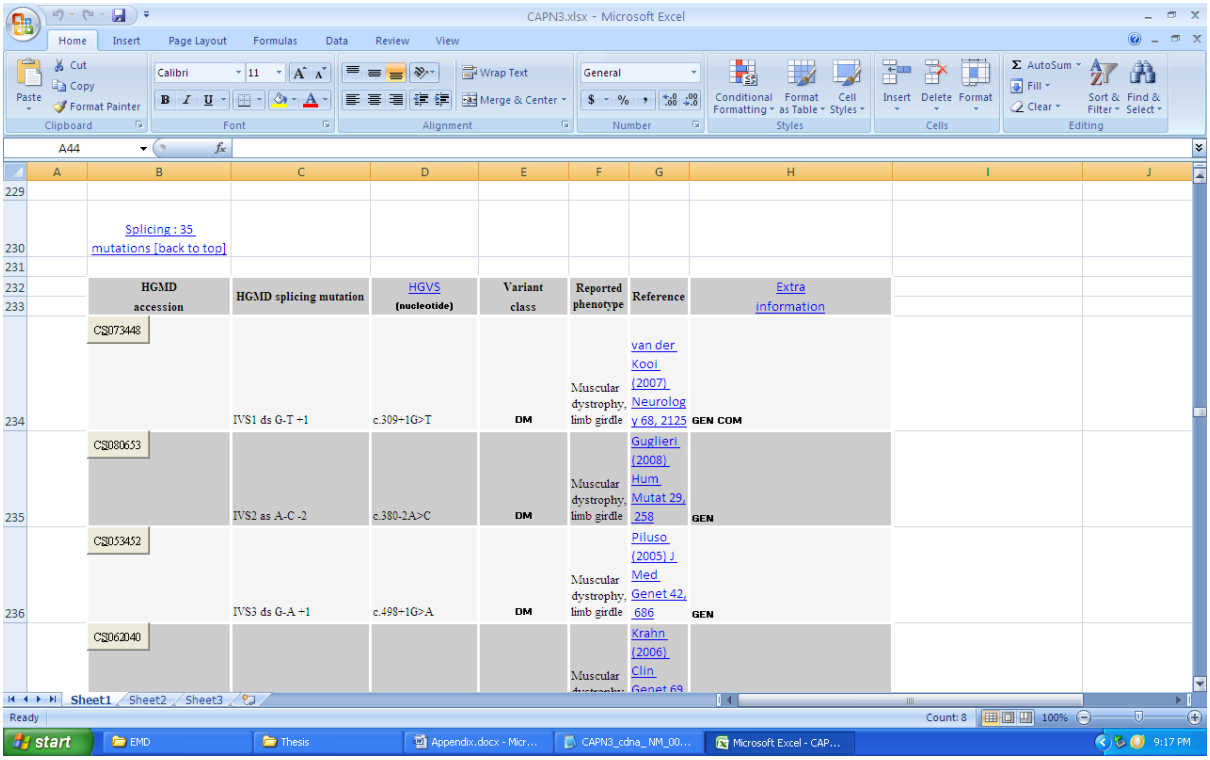
>gi|238018044|ref|NM_004006.2| Homo sapiens dystrophin (DMD), transcript variant Dp427m, mRNA
ATGTTGATACCACCTATCCAGATAAGAAGTCCATCTTAATGTACATCACATCACTCTTCC
AAGTTTTGCCTCAACAAGTGAGCATTGAAGCCATCCAGGAAGTGGAAATGTTGCCAAGGC
CACCTAAAGTGACTAAAGAAGAACATTTTCAGTTACATCATCAAATGCACTATTCTCAAC
AGATCACGGTCAGTCTAGCACAGGGATATGAGAGAACTTCTTCCCCTAAGCCTCGATTCA
AGAGCTATGCCTACACACAGGCTGCTTATGTCACCACCTCTGACCCTACACGGAGCCCAT
TTCCTTCACAGCATTTGGAAGCTCCTGAAGACAAGTCATTTGGCAGTTCATTGATGGAGA
GTGAAGTAAACCTGGACCGTTATCAAACAGCTTTAGAAGAAGTATTATCGTGGCTTCTTT
CTGCTGAGGACACATTGCAAGCACAAGGAGAGATTTCTAATGATGTGGAAGTGGTGAAAG
ACCAGTTTCATACTCATGAGGGGTACATGATGGATTTGACAGCCCATCAGGGCCGGGTTG
GTAATATTCTACAATTGGGAAGTAAGCTGATTGGAACAGGAAAATTATCAGAAGATGAAG
AAACTGAAGTACAAGAGCAGATGAATCTCCTAAATTCAAGATGGGAATGCCTCAGGGTAG
CTAGCATGGAAAAACAAAGCAATTTACATAGAGTTTTAATGGATCTCCAGAATCAGAAAC
TGAAAGAGTTGAATGACTGGCTAACAAAAACAGAAGAAAGAACAAGGAAAATGGAGGAAG
AGCCTCTTGGACCTGATCTTGAAGACCTAAAACGCCAAGTACAACAACATAAGGTGCTTC
AAGAAGATCTAGAACAAGAACAAGTCAGGGTCAATTCTCTCACTCACATGGTGGTGGTAG
TTGATGAATCTAGTGGAGATCACGCAACTGCTGCTTTGGAAGAACAACTTAAGGTATTGG
GAGATCGATGGGCAAACATCTGTAGATGGACAGAAGACCGCTGGGTTCTTTTACAAGACA
TCCTTCTCAAATGGCAACGTCTTACTGAAGAACAGTGCCTTTTTAGTGCATGGCTTTCAG
AAAAAGAAGATGCAGTGAACAAGATTCACACAACTGGCTTTAAAGATCAAAATGAAATGT
TATCAAGTCTTCAAAAACTGGCCGTTTTAAAAGCGGATCTAGAAAAGAAAAAGCAATCCA
TGGGCAAACTGTATTCACTCAAACAAGATCTTCTTTCAACACTGAAGAATAAGTCAGTGA
CCCAGAAGACGGAAGCATGGCTGGATAACTTTGCCCGGTGTTGGGATAATTTAGTCCAAA
AACTTGAAAAGAGTACAGCACAGATTTCACAGGCTGTCACCACCACTCAGCCATCACTAA
CACAGACAACTGTAATGGAAACAGTAACTACGGTGACCACAAGGGAACAGATCCTGGTAA

cDNA sequence for CAPN3 gene

>gi|27765081|ref|NM_000070.2| Homo sapiens calpain 3, (p94) (CAPN3), transcript variant 1, mRNA

ATGCCGACCGTCATTAGCGCATCTGTGGCTCCAAGGACAGCGGCTGAGCCCCGGTCCCCAGGGCCAGTTCCTCA
CCCGGCCCAGAGCAAGGCCACTGAGGCTGGGGGTGGAAACCCAAGTGGCATCTATTCAGCCATCATCAGCCGC
AATTTTCCTATTATCGGAGTGAAAGAGAAGACATTCGAGCAACTTCACAAGAAATGTCTAGAAAAGAAAGTTCT
TTATGTGGACCCTGAGTTCCCACCGGATGAGACCTCTCTCTTTTATAGCCAGAAGTTCCCCATCCAGTTCGTCTG
GAAGAGACCTCCGGAAATTTGCGAGAATCCCCGATTTATCATTGATGGAGCCAACAGAACTGACATCTGTCAAG
GAGAGCTAGGGGACTGCTGGTTTCTCGCAGCCATTGCCTGCCTGACCCTGAACCAGCACCTTCTTTTCCGAGTCA
TACCCCATGATCAAAGTTTCATCGAAAACTACGCAGGGATCTTCCACTTCCAGTTCTGGCGCTATGGAGAGTGG
GTGGACGTGGTTATAGATGACTGCCTGCCAACGTACAACAATCAACTGGTTTTCACCAAGTCCAACCACCGCAA
TGAGTTCTGGAGTGCTCTGCTGGAGAAGGCTTATGCTAAGCTCCATGGTTCCTACGAAGCTCTGAAAGGTGGGA
ACACCACAGAGGCCATGGAGGACTTCACAGGAGGGGTGGCAGAGTTTTTTGAGATCAGGGATGCTCCTAGTGAC
ATGTACAAGATCATGAAGAAAGCCATCGAGAGAGGCTCCCTCATGGGCTGCTCCATTGATGATGGCACGAACAT
GACCTATGGAACCTCTCCTTCTGGTCTGAACATGGGGGAGTTGATTGCACGGATGGTAAGGAATATGGATAACT
CACTGCTCCAGGACTCAGACCTCGACCCCAGAGGCTCAGATGAAAGACCGACCCGGACAATCATTCCGGTTCAG
TATGAGACAAGAATGGCCTGCGGGCTGGTCAGAGGTCACGCCTACTCTGTCACGGGGCTGGATGAGGTCCCGTT
CAAAGGTGAGAAAGTGAAGCTGGTGCGGCTGCGGAATCCGTGGGGCCAGGTGGAGTGGAACGGTTCTTGGAGT
GATAGATGGAAGGACTGGAGCTTTGTGGACAAAGATGAGAAGGCCCGTCTGCAGCACCAGGTCACTGAGGATG
GAGAGTTCTGGATGTCCTATGAGGATTTCATCTACCATTTCACAAAGTTGGAGATCTGCAACCTCACGGCCGATG
CTCTGCAGTCTGACAAGCTTCAGACCTGGACAGTGTCTGTGAACGAGGGCCGCTGGGTACGGGGTTGCTCTGCC
GGAGGCTGCCGCAACTTCCCAGATACTTTCTGGACCAACCCTCAGTACCGTCTGAAGCTCCTGGAGGAGGACGA
TGACCCTGATGACTCGGAGGTGATTTGCAGCTTCCTGGTGGCCCTGATGCAGAAGAACCGGCGGAAGGACCGGA
AGCTAGGGGCCAGTCTCTTCACCATTGGCTTCGCCATCTACGAGGTTCCCAAAGAGATGCACGGGAACAAGCAG
CACCTGCAGAAGGACTTCTTCCTGTACAACGCCTCCAAGGCCAGGAGCAAAACCTACATCAACATGCGGGAGGT
GTCCCAGCGCTTCCGCCTGCCTCCCAGCGAGTACGTCATCGTGCCCTCCACCTACGAGCCCCACCAGGAGGGGG
AATTCATCCTCCGGGTCTTCTCTGAAAAGAGGAACCTCTCTGAGGAAGTTGAAAATACCATCTCCGTGGATCGG
CCAGTGAAAAAGAAAAAAACCAAGCCCATCATCTTCGTTTCGGACAGAGCAAACAGCAACAAGGAGCTGGGTG
TGGACCAGGAGTCAGAGGAGGGCAAAGGCAAAACAAGCCCTGATAAGCAAAAGCAGTCCCCACAGCCACAGC
CTGGCAGCTCTGATCAGGAAAGTGAGGAACAGCAACAATTCCGGAACATTTTCAAGCAGATAGCAGGAGATGA
CATGGAGATCTGTGCAGATGAGCTCAAGAAGGTCCTTAACACAGTCGTGAACAAACACAAGGACCTGAAGACA
CACGGGTTCACACTGGAGTCCTGCCGTAGCATGATTGCGCTCATGGATACAGATGGCTCTGGAAAGCTCAACCT
GCAGGAGTTCCACCACCTCTGGAACAAGATTAAGGCCTGGCAGAAAATTTTCAAACACTATGACACAGACCAGT
CCGGCACCATCAACAGCTACGAGATGCGAAATGCAGTCAACGACGCAGGATTCCACCTCAACAACCAGCTCTAT
GACATCATTACCATGCGGTACGCAGACAAACACATGAACATCGACTTTGACAGTTTCATCTGCTGCTTCGTTAGG
CTGGAGGGCATGTTCAGAGCTTTTCATGCATTTGACAAGGATGGAGATGGTATCATCAAGCTCAACGTTCTGGA
GTGGCTGCAGCTCACCATGTATGCCTGA

Splicing mutation information for CAPN3 gene in HGMD



Splicing : 35 mutations [back to top]

HGMD accession	HGMD splicing mutation	HGVS (nucleotide)	Variant class	Reported phenotype	Reference	Extra information
CS073448	IVS1 ds G-T +1	c.309+1G>T	DM	Muscular dystrophy, limb girdle	van der Kooi (2007) Neurology 68, 2125	GEN COM
CS080653	IVS2 as A-C -2	c.380-2A>C	DM	Muscular dystrophy, limb girdle	Guglieri (2008) Hum Mutat 29, 258	GEN
CS053452	IVS3 ds G-A +1	c.498+1G>A	DM	Muscular dystrophy, limb girdle	Piluso (2005) J Med Genet 42, 686	GEN
CS062040				Muscular	Krahn (2006) Clin Genet 69	

Splicing mutated gene sequence for CAPN3 – CS080653

>gi|27765081|ref|NM_000070.2| Homo sapiens calpain 3, (p94) (CAPN3), transcript variant 1, mRNA

ATGCCGACCGTCATTAGCGCATCTGTGGCTCCAAGGACAGCGGCTGAGCCCCGGTCCCCAGGGCCAGTTCCTCACCCGGC
CCAGAGCAAGGCCACTGAGGCTGGGGGTGGAAACCCAAGTGGCATCTATTCAGCCATCATCAGCCGCAATTTTCCTATTA
TCGGAGTGAAAGAGAAGACATTCGAGCAACTTCACAAGAAATGTCTAGAAAAGAAAGTTCTTTATGTGGACCCTGAGTT
CCCACCGGATGAGACCTCTCTCTTTTATAGCCAGAAGTTCCCCATCCAGTTCGTCTGGAAGAGACCTCCGGAAATTTGCG
AGAATCCCCGATTTATCATTGATGGAGCCAACAGAACTGACATCTGTCAAGGAGAGCTAGGGGACTGCTGGTTTCTCGCA
GCCATTGCCTGCCTGACCCTGAACCAGCACCTTCTTTTCCGAGTCATACCCCATGATCAAAGTTTCATCGAAAACTACGCA
GGGATCTTCCACTTCCAGTTCTGGCGCTATGGAGAGTGGGTGGACGTGGTTATAGATGACTGCCTGCCAACGTACAACAA
TCAACTGGTTTTCACCAAGTCCAACCACCGCAATGAGTTCTGGAGTGCTCTGCTGGAGAAGGCTTATGCTAAGCTCCATG
GTTCCTACGAAGCTCTGAAAGGTGGGAACACCACAGAGGCCATGGAGGACTTCACAGGAGGGGTGGCAGAGTTTTTTGA
GATCAGGGATGCTCCTAGTGACATGTACAAGATCATGAAGAAAGCCATCGAGAGAGGCTCCCTCATGGGCTGCTCCATTG
ATGATGGCACGAACATGACCTATGGAACCTCTCCTTCTGGTCTGAACATGGGGGAGTTGATTGCACGGATGGTAAGGAAT
ATGGATAACTCACTGCTCCAGGACTCAGACCTCGACCCCAGAGGCTCAGATGAAAGACCGACCCGGACAATCATTCCGG
TTCAGTATGAGACAAGAATGGCCTGCGGGCTGGTCAGAGGTCACGCCTACTCTGTCACGGGGCTGGATGAGGTCCCGTTC
AAAGGTGAGAAAGTGAAGCTGGTGCGGCTGCGGAATCCGTGGGGCCAGGTGGAGTGGAACGGTTCTTGGAGTGATAGAT
GGAAGGACTGGAGCTTTGTGGACAAAGATGAGAAGGCCCGTCTGCAGCACCAGGTCACTGAGGATGGAGAGTTCTGGAT
GTCCTATGAGGATTTCATCTACCATTTCACAAAGTTGGAGATCTGCAACCTCACGGCCGATGCTCTGCAGTCTGACAAGCT
TCAGACCTGGACAGTGTCTGTGAACGAGGGCCGCTGGGTACGGGGTTGCTCTGCCGGAGGCTGCCGCAACTTCCCAGATA
CTTTCTGGACCAACCCTCAGTACCGTCTGAAGCTCCTGGAGGAGGACGATGACCCTGATGACTCGGAGGTGATTTGCAGC
TTCCTGGTGGCCCTGATGCAGAAGAACCGGCGGAAGGACCGGAAGCTAGGGGCCAGTCTCTTCACCATTGGCTTCGCCAT
CTACGAGGTTCCCAAAGAGATGCACGGGAACAAGCAGCACCTGCAGAAGGACTTCTTCCTGTACAACGCCTCCAAGGCC
AGGAGCAAAACCTACATCAACATGCGGGAGGTGTCCCAGCGCTTCCGCCTGCCTCCCAGCGAGTACGTCATCGTGCCCTC
CACCTACGAGCCCCACCAGGAGGGGGAATTCATCCTCCGGGTCTTCTCTGAAAAGAGGAACCTCTCTGAGGAAGTTGAA
AATACCATCTCCGTGGATCGGCCAGTGAAAAAGAAAAAAACCAAGCCCATCATCTTCGTTTCGGACAGAGCAAACAGCA
ACAAGGAGCTGGGTGTGGACCAGGAGTCAGAGGAGGGCAAAGGCAAAACAAGCCCTGATAAGCAAAAGCAGTCCCCAC
AGCCACAGCCTGGCAGCTCTGATCAGGAAAGTGAGGAACAGCAACAATTCCGGAACATTTTCAAGCAGATAGCAGGAGA
TGACATGGAGATCTGTGCAGATGAGCTCAAGAAGGTCCTTAACACAGTCGTGAACAAACACAAGGACCTGAAGACACAC
GGGTTCACACTGGAGTCCTGCCGTAGCATGATTGCGCTCATGGATACAGATGGCTCTGGAAAGCTCAACCTGCAGGAGTT
CCACCACCTCTGGAACAAGATTAAGGCCTGGCAGAAAATTTTCAAACACTATGACACAGACCAGTCCGGCACCATCAAC
AGCTACGAGATGCGAAATGCAGTCAACGACGCAGGATTCCACCTCAACAACCAGCTCTATGACATCATTACCATGCGGTA
CGCAGACAAACACATGAACATCGACTTTGACAGTTTCATCTGCTGCTTCGTTAGGCTGGAGGGCATGTTCAGAGCTTTTC
ATGCATTTGACAAGGATGGAGATGGTATCATCAAGCTCAACGTTCTGGAGTGGCTGCAGCTCACCATGTATGCCTGA

Appendix – B

Feature Extraction and Feature vectors

R Script of Feature extraction

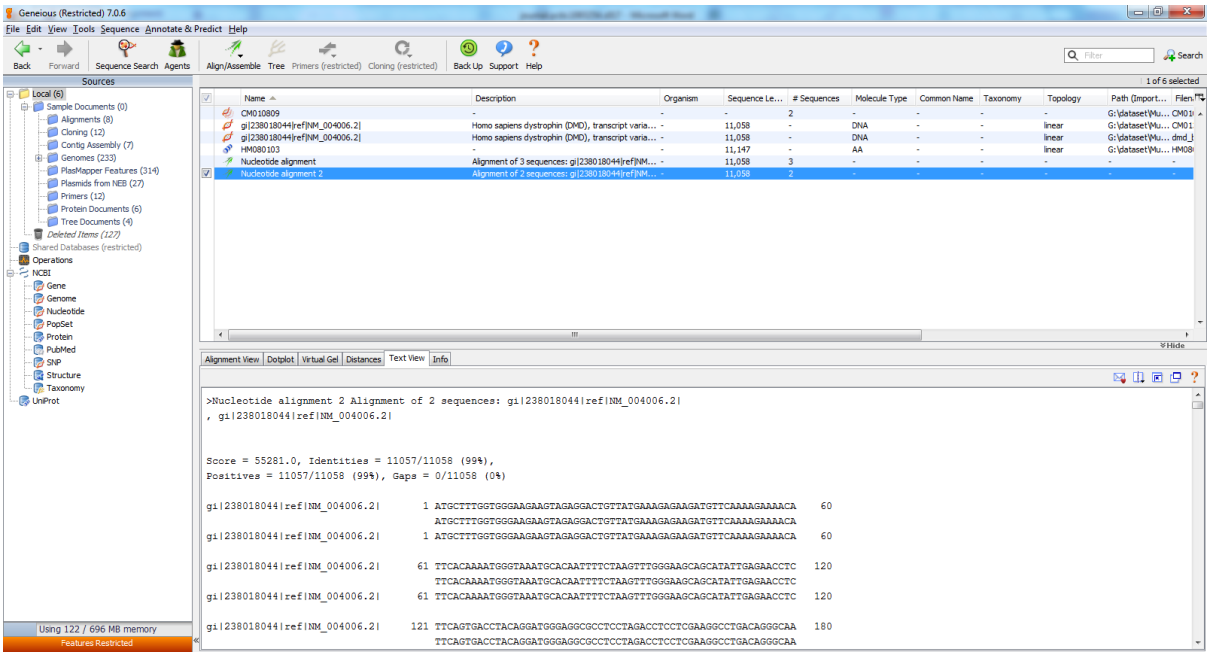
```
##########Retrieveing Gene id,Gene symbol and chromosome number
source("http://bioconductor.org/biocLite.R")
library(biomaRt)
listMarts()
ens <- useMart("ensembl")
listDatasets(ens)
ens <- useDataset("hsapiens_gene_ensembl",mart=ens)
getGene(id=2010, type="entrezgene", mart=ens)
############Creating a mutated sequence file
library(seqinr)
require(ade4)
library(Biostrings)
d1<-read.fasta(file = "SH3TC2_cdna_ NM_024577.3.fasta")
ds1 <- d1[[1]]
mp<-596
ds1[mp]
x=read.fasta("SH3TC2_cdna_ NM_024577.3.fasta")
new=lapply(seq(length(x)), function(i) {
    s2c(gsub("c","t",c2s(getSequence(x[[i]]))))
})
write.fasta(new,names=names(x),file="sample.fasta",nbchar=60)

d2<-read.fasta(file = "sample.fasta")
ds2 <- d2[[1]]
ds2[mp]
################ Length of the sequence
ln1 <- length(ds1)
ln2 <- length(ds2)
################ Mutated codon position
cod_p<-mp/3
print(mp)
print(ln2)
print(cod_p)
cod_pos<-round(cod_p)
```

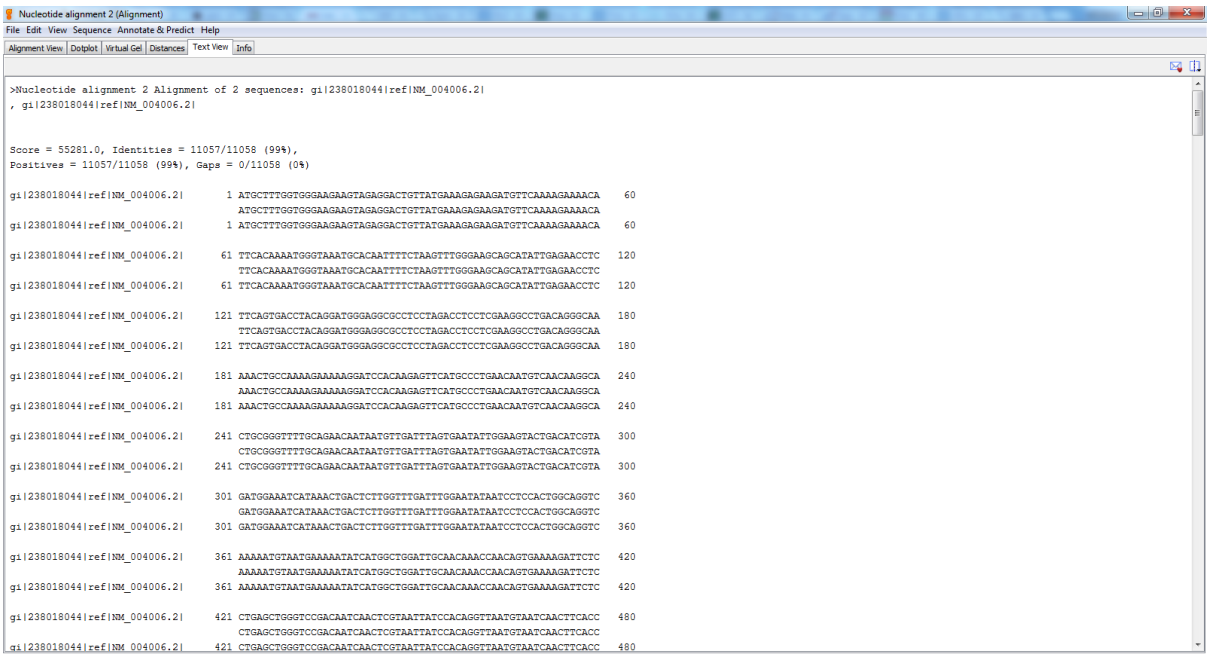
```
################ Splitting the sequence into codons
cod<-splitseq(ds1)
cod[199]
ori<-cod[cod_pos]
ori
cod<-splitseq(ds2)
mut<-cod[cod_pos]
mut
################ Observed allele
tablecode()
ala<-1
arg<-2
asn<-3
asp<-4
cys<-5
gln<-6
glu<-7
gly<-8
his<-9
ile<-10
leu<-11
lys<-12
met<-13
phe<-14
pro<-15
ser<-16
thr<-17
trp<-18
tyr<-19
val<-20
################## Reference allele
if ((mut == "tag") || (mut == "taa") || (mut == "tga")){
 ref<-0}
print(ref)
length(cod)
```

```
#################### Mutation start and mutation end
for (j in 1:length(cod)){
i<-cod[[j]]
c<-paste(j,i,sep="-")
cat(c,file="sample.fasta",sep="\t",append=TRUE)}
cod1<-read.table(file="out2.txt")
cod1<-read.fasta(file = "sample.fasta")
cod2 <-cod1 [[1]]
dstart<-ds[1:650]
length[dstart]
dstartstring <- c2s(dstart)
matchPattern("ata", dstartstring)
###################### Len variant, protein changed
lv<-ln1-ln2
if(ori == ref)
{lv<-3
}
if(ln1==ln2)
{if(ref == 0)
{lv<-1}
else
{lv<-2}}
if((lv == 1) || (lv == 2) || (lv == 4) || (lv == 5))
{ph<-1}
print(ph)
################################## Alteration Type
print(lv)
if(lv == 1){
altype<-1
}else if(lv == 2){
altype<-2
}else if(lv == 3){
altype<-3
}else if(lv<ln1){
altype<-4
}else if(lv>ln1){
altype<-5
}print(altype)
```

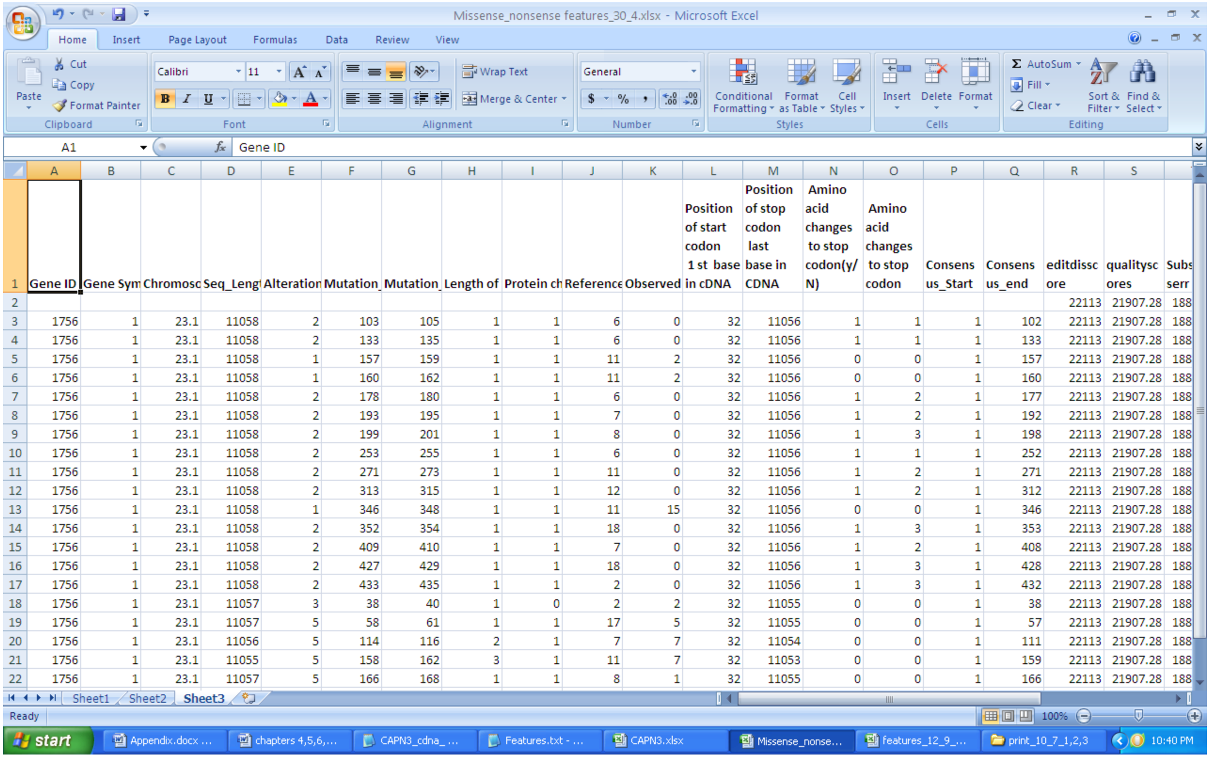
```
######################## Amino acid to stop codon, amino acid stop codon
if(ref == 0){
am_st<-1
am_type<-1
}else{
am_st<-1
am_type<-0
}
print(am_st)
print(am_type)
######################## Position of start and stop codon
tablecode()
ds <- d2[[1]]
length(ds)
dstartstring <- c2s(ds)
matchPattern("tag", dstartstring)
#################################### pairwise alignment
d1<-read.fasta("SH3TC2_cdna_ NM_024577.3.fasta")
d2<-read.fasta(file = "sample.fasta")
s1 <- toupper(c2s(d1[[1]]))
s2 <- toupper(c2s(d2[[1]]))
## Fit a global pairwise alignment using edit distance scoring
a1 <- pairwiseAlignment(s1, s2,substitutionMatrix = nucleotideSubstitutionMatrix(2, -1, TRUE),gapOpening = -2,
gapExtension = -8)
## Examine quality-based match and mismatch bit scores for DNA/RNA
```

Pairwise alignment scores - Geneiouspro Output



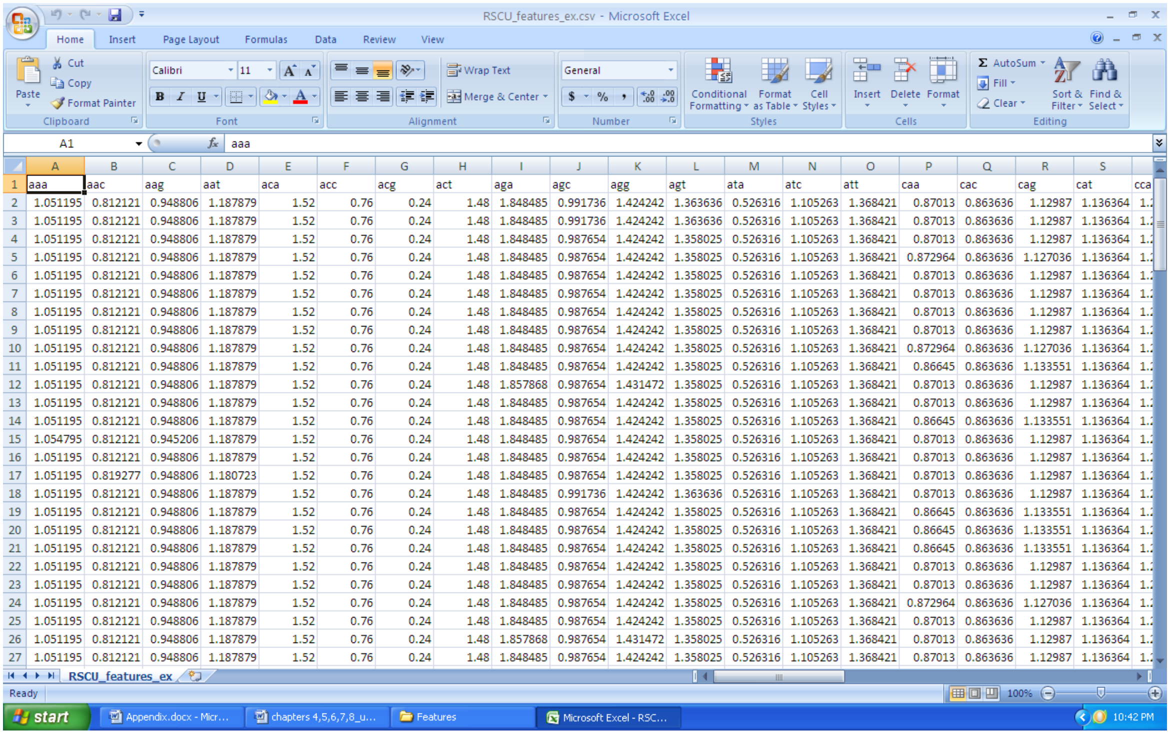


Feature vectors of Non – Synonymous Mutated Gene Sequences



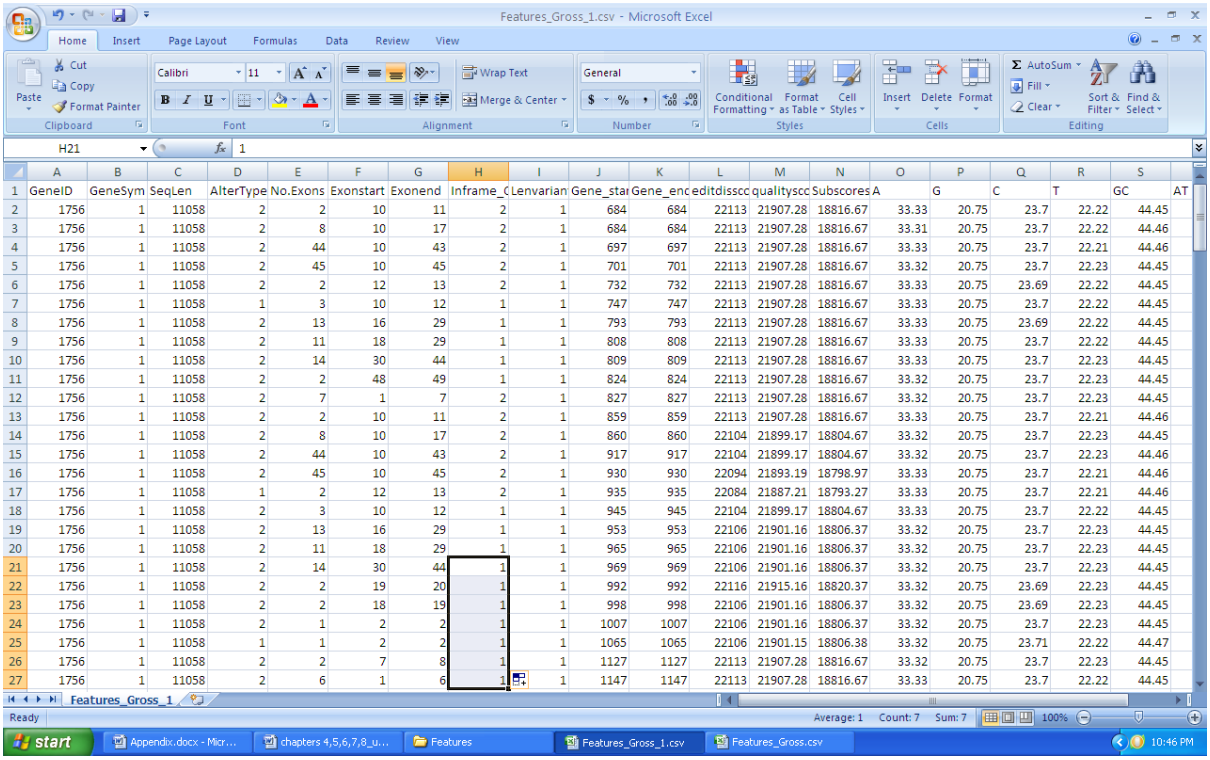
	Gene ID	Gene Sym	Chromoso	Seq_Lengt	Alteration	Mutation_	Mutation_	Length of	Protein ch	Reference	Observed	Position of start codon 1 st base in cDNA	Position of stop codon last base in CDNA	Amino acid changes to stop codon(y/N)	Amino acid changes to stop codon	Consensus_Start	Consensus_end	editdisscore	qualityscores	Subs serr
2																		22113	21907.28	188
3	1756	1	23.1	11058	2	103	105	1	1	6	0	32	11056	1	1	1	102	22113	21907.28	188
4	1756	1	23.1	11058	2	133	135	1	1	6	0	32	11056	1	1	1	133	22113	21907.28	188
5	1756	1	23.1	11058	1	157	159	1	1	11	2	32	11056	0	0	1	157	22113	21907.28	188
6	1756	1	23.1	11058	1	160	162	1	1	11	2	32	11056	0	0	1	160	22113	21907.28	188
7	1756	1	23.1	11058	2	178	180	1	1	6	0	32	11056	1	2	1	177	22113	21907.28	188
8	1756	1	23.1	11058	2	193	195	1	1	7	0	32	11056	1	2	1	192	22113	21907.28	188
9	1756	1	23.1	11058	2	199	201	1	1	8	0	32	11056	1	3	1	198	22113	21907.28	188
10	1756	1	23.1	11058	2	253	255	1	1	6	0	32	11056	1	1	1	252	22113	21907.28	188
11	1756	1	23.1	11058	2	271	273	1	1	11	0	32	11056	1	2	1	271	22113	21907.28	188
12	1756	1	23.1	11058	2	313	315	1	1	12	0	32	11056	1	2	1	312	22113	21907.28	188
13	1756	1	23.1	11058	1	346	348	1	1	11	15	32	11056	0	0	1	346	22113	21907.28	188
14	1756	1	23.1	11058	2	352	354	1	1	18	0	32	11056	1	3	1	353	22113	21907.28	188
15	1756	1	23.1	11058	2	409	410	1	1	7	0	32	11056	1	2	1	408	22113	21907.28	188
16	1756	1	23.1	11058	2	427	429	1	1	18	0	32	11056	1	3	1	428	22113	21907.28	188
17	1756	1	23.1	11058	2	433	435	1	1	2	0	32	11056	1	3	1	432	22113	21907.28	188
18	1756	1	23.1	11057	3	38	40	1	0	2	2	32	11055	0	0	1	38	22113	21907.28	188
19	1756	1	23.1	11057	5	58	61	1	1	17	5	32	11055	0	0	1	57	22113	21907.28	188
20	1756	1	23.1	11056	5	114	116	2	1	7	7	32	11054	0	0	1	111	22113	21907.28	188
21	1756	1	23.1	11055	5	158	162	3	1	11	7	32	11053	0	0	1	159	22113	21907.28	188
22	1756	1	23.1	11057	5	166	168	1	1	8	1	32	11055	0	0	1	166	22113	21907.28	188

Feature vectors of Synonymous Mutated Gene Sequences



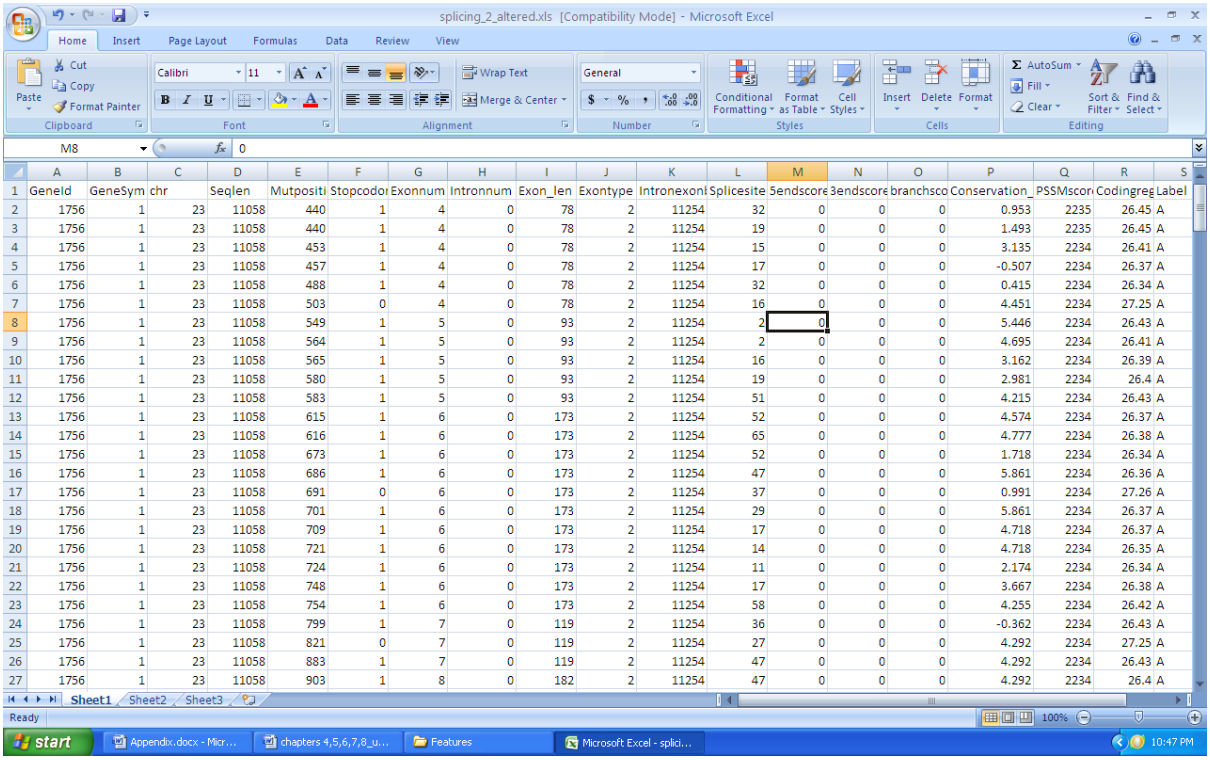
	aaa	aac	aag	aat	aca	acc	acg	act	aga	agc	agg	agt	ata	atc	att	caa	cac	cag	cat	cca
2	1.051195	0.812121	0.948806	1.187879	1.52	0.76	0.24	1.48	1.848485	0.991736	1.424242	1.363636	0.526316	1.105263	1.368421	0.87013	0.863636	1.12987	1.136364	1.
3	1.051195	0.812121	0.948806	1.187879	1.52	0.76	0.24	1.48	1.848485	0.991736	1.424242	1.363636	0.526316	1.105263	1.368421	0.87013	0.863636	1.12987	1.136364	1.
4	1.051195	0.812121	0.948806	1.187879	1.52	0.76	0.24	1.48	1.848485	0.987654	1.424242	1.358025	0.526316	1.105263	1.368421	0.87013	0.863636	1.12987	1.136364	1.
5	1.051195	0.812121	0.948806	1.187879	1.52	0.76	0.24	1.48	1.848485	0.987654	1.424242	1.358025	0.526316	1.105263	1.368421	0.872964	0.863636	1.127036	1.136364	1.
6	1.051195	0.812121	0.948806	1.187879	1.52	0.76	0.24	1.48	1.848485	0.987654	1.424242	1.358025	0.526316	1.105263	1.368421	0.87013	0.863636	1.12987	1.136364	1.
7	1.051195	0.812121	0.948806	1.187879	1.52	0.76	0.24	1.48	1.848485	0.987654	1.424242	1.358025	0.526316	1.105263	1.368421	0.87013	0.863636	1.12987	1.136364	1.
8	1.051195	0.812121	0.948806	1.187879	1.52	0.76	0.24	1.48	1.848485	0.987654	1.424242	1.358025	0.526316	1.105263	1.368421	0.87013	0.863636	1.12987	1.136364	1.
9	1.051195	0.812121	0.948806	1.187879	1.52	0.76	0.24	1.48	1.848485	0.987654	1.424242	1.358025	0.526316	1.105263	1.368421	0.87013	0.863636	1.12987	1.136364	1.
10	1.051195	0.812121	0.948806	1.187879	1.52	0.76	0.24	1.48	1.848485	0.987654	1.424242	1.358025	0.526316	1.105263	1.368421	0.872964	0.863636	1.127036	1.136364	1.
11	1.051195	0.812121	0.948806	1.187879	1.52	0.76	0.24	1.48	1.848485	0.987654	1.424242	1.358025	0.526316	1.105263	1.368421	0.86645	0.863636	1.133551	1.136364	1.
12	1.051195	0.812121	0.948806	1.187879	1.52	0.76	0.24	1.48	1.857868	0.987654	1.431472	1.358025	0.526316	1.105263	1.368421	0.87013	0.863636	1.12987	1.136364	1.
13	1.051195	0.812121	0.948806	1.187879	1.52	0.76	0.24	1.48	1.848485	0.987654	1.424242	1.358025	0.526316	1.105263	1.368421	0.87013	0.863636	1.12987	1.136364	1.
14	1.051195	0.812121	0.948806	1.187879	1.52	0.76	0.24	1.48	1.848485	0.987654	1.424242	1.358025	0.526316	1.105263	1.368421	0.86645	0.863636	1.133551	1.136364	1.
15	1.054795	0.812121	0.945206	1.187879	1.52	0.76	0.24	1.48	1.848485	0.987654	1.424242	1.358025	0.526316	1.105263	1.368421	0.87013	0.863636	1.12987	1.136364	1.
16	1.051195	0.812121	0.948806	1.187879	1.52	0.76	0.24	1.48	1.848485	0.987654	1.424242	1.358025	0.526316	1.105263	1.368421	0.87013	0.863636	1.12987	1.136364	1.
17	1.051195	0.819277	0.948806	1.180723	1.52	0.76	0.24	1.48	1.848485	0.987654	1.424242	1.358025	0.526316	1.105263	1.368421	0.87013	0.863636	1.12987	1.136364	1.
18	1.051195	0.812121	0.948806	1.187879	1.52	0.76	0.24	1.48	1.848485	0.991736	1.424242	1.363636	0.526316	1.105263	1.368421	0.87013	0.863636	1.12987	1.136364	1.
19	1.051195	0.812121	0.948806	1.187879	1.52	0.76	0.24	1.48	1.848485	0.987654	1.424242	1.358025	0.526316	1.105263	1.368421	0.86645	0.863636	1.133551	1.136364	1.
20	1.051195	0.812121	0.948806	1.187879	1.52	0.76	0.24	1.48	1.848485	0.987654	1.424242	1.358025	0.526316	1.105263	1.368421	0.86645	0.863636	1.133551	1.136364	1.
21	1.051195	0.812121	0.948806	1.187879	1.52	0.76	0.24	1.48	1.848485	0.987654	1.424242	1.358025	0.526316	1.105263	1.368421	0.86645	0.863636	1.133551	1.136364	1.
22	1.051195	0.812121	0.948806	1.187879	1.52	0.76	0.24	1.48	1.848485	0.987654	1.424242	1.358025	0.526316	1.105263	1.368421	0.87013	0.863636	1.12987	1.136364	1.
23	1.051195	0.812121	0.948806	1.187879	1.52	0.76	0.24	1.48	1.848485	0.987654	1.424242	1.358025	0.526316	1.105263	1.368421	0.87013	0.863636	1.12987	1.136364	1.
24	1.051195	0.812121	0.948806	1.187879	1.52	0.76	0.24	1.48	1.848485	0.987654	1.424242	1.358025	0.526316	1.105263	1.368421	0.872964	0.863636	1.127036	1.136364	1.
25	1.051195	0.812121	0.948806	1.187879	1.52	0.76	0.24	1.48	1.848485	0.987654	1.424242	1.358025	0.526316	1.105263	1.368421	0.87013	0.863636	1.12987	1.136364	1.
26	1.051195	0.812121	0.948806	1.187879	1.52	0.76	0.24	1.48	1.857868	0.987654	1.431472	1.358025	0.526316	1.105263	1.368421	0.87013	0.863636	1.12987	1.136364	1.
27	1.051195	0.812121	0.948806	1.187879	1.52	0.76	0.24	1.48	1.848485	0.987654	1.424242	1.358025	0.526316	1.105263	1.368421	0.87013	0.863636	1.12987	1.136364	1.

Feature vectors of Insertion, Deletion Mutated Gene Sequences



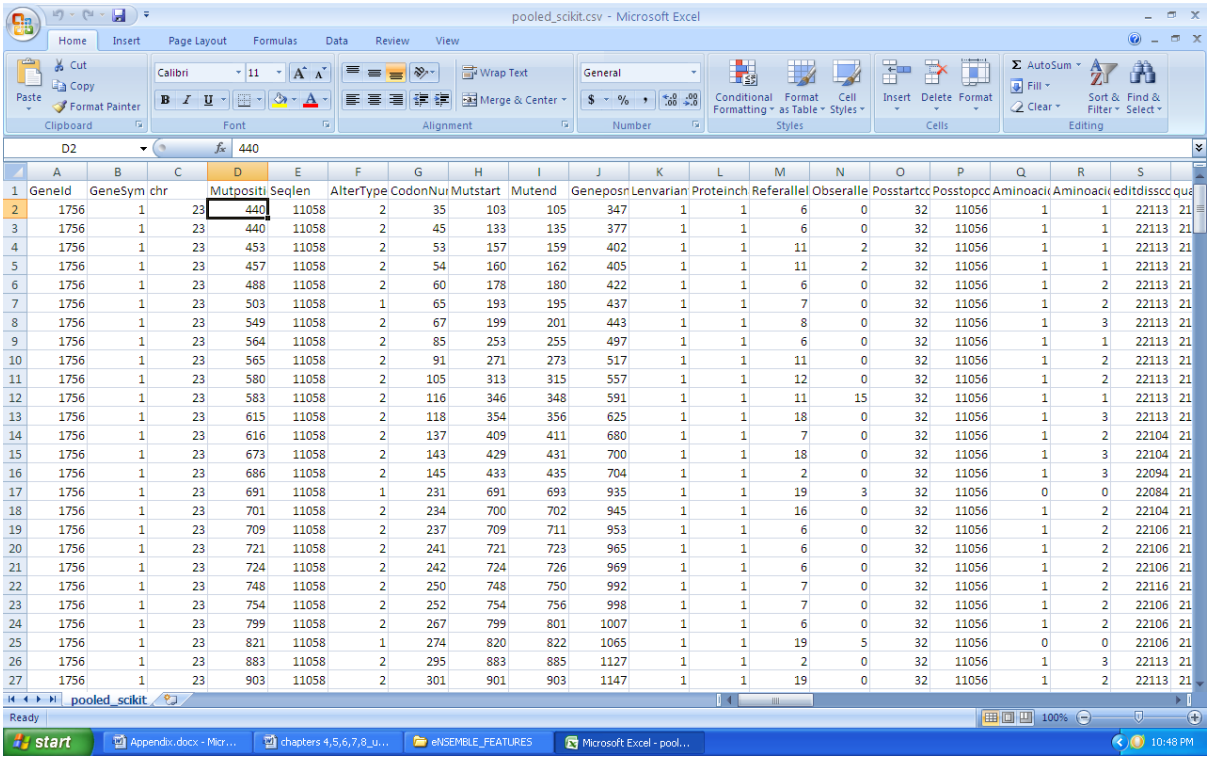
	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S
1	GeneID	GeneSym	SeqLen	AlterType	No.Exons	Exonstart	Exonend	Inframe_C	Lenvarian	Gene_star	Gene_enc	editdissc	qualityscc	Subscores	A	G	C	T	GC	AT
2	1756	1	11058	2	2	10	11	2	1	684	684	22113	21907.28	18816.67	33.33	20.75	23.7	22.22	44.45	
3	1756	1	11058	2	8	10	17	2	1	684	684	22113	21907.28	18816.67	33.31	20.75	23.7	22.22	44.46	
4	1756	1	11058	2	44	10	43	2	1	697	697	22113	21907.28	18816.67	33.33	20.75	23.7	22.21	44.46	
5	1756	1	11058	2	45	10	45	2	1	701	701	22113	21907.28	18816.67	33.32	20.75	23.7	22.23	44.45	
6	1756	1	11058	2	2	12	13	2	1	732	732	22113	21907.28	18816.67	33.33	20.75	23.69	22.22	44.45	
7	1756	1	11058	1	3	10	12	1	1	747	747	22113	21907.28	18816.67	33.33	20.75	23.7	22.22	44.45	
8	1756	1	11058	2	13	16	29	1	1	793	793	22113	21907.28	18816.67	33.33	20.75	23.69	22.22	44.45	
9	1756	1	11058	2	11	18	29	1	1	808	808	22113	21907.28	18816.67	33.33	20.75	23.7	22.22	44.45	
10	1756	1	11058	2	14	30	44	1	1	809	809	22113	21907.28	18816.67	33.33	20.75	23.7	22.23	44.45	
11	1756	1	11058	2	2	48	49	1	1	824	824	22113	21907.28	18816.67	33.32	20.75	23.7	22.23	44.45	
12	1756	1	11058	2	7	1	7	2	1	827	827	22113	21907.28	18816.67	33.32	20.75	23.7	22.23	44.45	
13	1756	1	11058	2	2	10	11	2	1	859	859	22113	21907.28	18816.67	33.33	20.75	23.7	22.21	44.46	
14	1756	1	11058	2	8	10	17	2	1	860	860	22104	21899.17	18804.67	33.32	20.75	23.7	22.23	44.45	
15	1756	1	11058	2	44	10	43	2	1	917	917	22104	21899.17	18804.67	33.32	20.75	23.7	22.23	44.46	
16	1756	1	11058	2	45	10	45	2	1	930	930	22094	21893.19	18798.97	33.33	20.75	23.7	22.21	44.46	
17	1756	1	11058	1	2	12	13	2	1	935	935	22084	21887.21	18793.27	33.33	20.75	23.7	22.21	44.46	
18	1756	1	11058	2	3	10	12	1	1	945	945	22104	21899.17	18804.67	33.33	20.75	23.7	22.22	44.45	
19	1756	1	11058	2	13	16	29	1	1	953	953	22106	21901.16	18806.37	33.32	20.75	23.7	22.23	44.45	
20	1756	1	11058	2	11	18	29	1	1	965	965	22106	21901.16	18806.37	33.32	20.75	23.7	22.23	44.45	
21	1756	1	11058	2	14	30	44	1	1	969	969	22106	21901.16	18806.37	33.32	20.75	23.7	22.23	44.45	
22	1756	1	11058	2	2	19	20	1	1	992	992	22116	21915.16	18820.37	33.32	20.75	23.69	22.23	44.45	
23	1756	1	11058	2	2	18	19	1	1	998	998	22106	21901.16	18806.37	33.32	20.75	23.69	22.23	44.45	
24	1756	1	11058	2	1	2	2	1	1	1007	1007	22106	21901.16	18806.37	33.32	20.75	23.7	22.23	44.45	
25	1756	1	11058	1	1	2	2	1	1	1065	1065	22106	21901.15	18806.38	33.32	20.75	23.71	22.22	44.47	
26	1756	1	11058	2	2	7	8	1	1	1127	1127	22113	21907.28	18816.67	33.32	20.75	23.7	22.23	44.45	
27	1756	1	11058	2	6	1	6	1	1	1147	1147	22113	21907.28	18816.67	33.33	20.75	23.7	22.22	44.45	

Feature vectors of Splicing Mutated Gene Sequences



	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S
1	GeneId	GeneSym	chr	Seqlen	Mutpositi	Stopcodor	Exonnum	Intronnum	Exon_len	Exontype	Intronexonl	Splicesite	Sendscore	Sendscore	branchsco	Conservation_	PSSMscor	Codingreg	Label
2	1756	1	23	11058	440	1	4	0	78	2	11254	32	0	0	0	0.953	2235	26.45	A
3	1756	1	23	11058	440	1	4	0	78	2	11254	19	0	0	0	1.493	2235	26.45	A
4	1756	1	23	11058	453	1	4	0	78	2	11254	15	0	0	0	3.135	2234	26.41	A
5	1756	1	23	11058	457	1	4	0	78	2	11254	17	0	0	0	-0.507	2234	26.37	A
6	1756	1	23	11058	488	1	4	0	78	2	11254	32	0	0	0	0.415	2234	26.34	A
7	1756	1	23	11058	503	0	4	0	78	2	11254	16	0	0	0	4.451	2234	27.25	A
8	1756	1	23	11058	549	1	5	0	93	2	11254	2	0	0	0	5.446	2234	26.43	A
9	1756	1	23	11058	564	1	5	0	93	2	11254	2	0	0	0	4.695	2234	26.41	A
10	1756	1	23	11058	565	1	5	0	93	2	11254	16	0	0	0	3.162	2234	26.39	A
11	1756	1	23	11058	580	1	5	0	93	2	11254	19	0	0	0	2.981	2234	26.4	A
12	1756	1	23	11058	583	1	5	0	93	2	11254	51	0	0	0	4.215	2234	26.43	A
13	1756	1	23	11058	615	1	6	0	173	2	11254	52	0	0	0	4.574	2234	26.37	A
14	1756	1	23	11058	616	1	6	0	173	2	11254	65	0	0	0	4.777	2234	26.38	A
15	1756	1	23	11058	673	1	6	0	173	2	11254	52	0	0	0	1.718	2234	26.34	A
16	1756	1	23	11058	686	1	6	0	173	2	11254	47	0	0	0	5.861	2234	26.36	A
17	1756	1	23	11058	691	0	6	0	173	2	11254	37	0	0	0	0.991	2234	27.26	A
18	1756	1	23	11058	701	1	6	0	173	2	11254	29	0	0	0	5.861	2234	26.37	A
19	1756	1	23	11058	709	1	6	0	173	2	11254	17	0	0	0	4.718	2234	26.37	A
20	1756	1	23	11058	721	1	6	0	173	2	11254	14	0	0	0	4.718	2234	26.35	A
21	1756	1	23	11058	724	1	6	0	173	2	11254	11	0	0	0	2.174	2234	26.34	A
22	1756	1	23	11058	748	1	6	0	173	2	11254	17	0	0	0	3.667	2234	26.38	A
23	1756	1	23	11058	754	1	6	0	173	2	11254	58	0	0	0	4.255	2234	26.42	A
24	1756	1	23	11058	799	1	7	0	119	2	11254	36	0	0	0	-0.362	2234	26.43	A
25	1756	1	23	11058	821	0	7	0	119	2	11254	27	0	0	0	4.292	2234	27.25	A
26	1756	1	23	11058	883	1	7	0	119	2	11254	47	0	0	0	4.292	2234	26.43	A
27	1756	1	23	11058	903	1	8	0	182	2	11254	47	0	0	0	4.292	2234	26.4	A

Feature vectors of AGM dataset



	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	
1	GeneId	GeneSym	chr	Mutpositi	Seqlen	AlterType	CodonNur	Mutstart	Mutend	Geneposn	Lenvarian	Proteinch	Referallel	Obseralle	Posstartcc	Posstopcc	Aminoacic	Aminoacic	editdissc	qua
2	1756	1	23	440	11058	2	35	103	105	347	1	1	6	0	32	11056	1	1	22113	21
3	1756	1	23	440	11058	2	45	133	135	377	1	1	6	0	32	11056	1	1	22113	21
4	1756	1	23	453	11058	2	53	157	159	402	1	1	11	2	32	11056	1	1	22113	21
5	1756	1	23	457	11058	2	54	160	162	405	1	1	11	2	32	11056	1	1	22113	21
6	1756	1	23	488	11058	2	60	178	180	422	1	1	6	0	32	11056	1	2	22113	21
7	1756	1	23	503	11058	1	65	193	195	437	1	1	7	0	32	11056	1	2	22113	21
8	1756	1	23	549	11058	2	67	199	201	443	1	1	8	0	32	11056	1	3	22113	21
9	1756	1	23	564	11058	2	85	253	255	497	1	1	6	0	32	11056	1	1	22113	21
10	1756	1	23	565	11058	2	91	271	273	517	1	1	11	0	32	11056	1	2	22113	21
11	1756	1	23	580	11058	2	105	313	315	557	1	1	12	0	32	11056	1	2	22113	21
12	1756	1	23	583	11058	2	116	346	348	591	1	1	11	15	32	11056	1	1	22113	21
13	1756	1	23	615	11058	2	118	354	356	625	1	1	18	0	32	11056	1	3	22113	21
14	1756	1	23	616	11058	2	137	409	411	680	1	1	7	0	32	11056	1	2	22104	21
15	1756	1	23	673	11058	2	143	429	431	700	1	1	18	0	32	11056	1	3	22104	21
16	1756	1	23	686	11058	2	145	433	435	704	1	1	2	0	32	11056	1	3	22094	21
17	1756	1	23	691	11058	1	231	691	693	935	1	1	19	3	32	11056	0	0	22084	21
18	1756	1	23	701	11058	2	234	700	702	945	1	1	16	0	32	11056	1	2	22104	21
19	1756	1	23	709	11058	2	237	709	711	953	1	1	6	0	32	11056	1	2	22106	21
20	1756	1	23	721	11058	2	241	721	723	965	1	1	6	0	32	11056	1	2	22106	21
21	1756	1	23	724	11058	2	242	724	726	969	1	1	6	0	32	11056	1	2	22106	21
22	1756	1	23	748	11058	2	250	748	750	992	1	1	7	0	32	11056	1	2	22116	21
23	1756	1	23	754	11058	2	252	754	756	998	1	1	7	0	32	11056	1	2	22106	21
24	1756	1	23	799	11058	2	267	799	801	1007	1	1	6	0	32	11056	1	2	22106	21
25	1756	1	23	821	11058	1	274	820	822	1065	1	1	19	5	32	11056	0	0	22106	21
26	1756	1	23	883	11058	2	295	883	885	1127	1	1	2	0	32	11056	1	3	22113	21
27	1756	1	23	903	11058	2	301	901	903	1147	1	1	19	0	32	11056	1	2	22113	21

Appendix – C

Python Code

Naïve Bayes Classifier

```
import numpy as np
import io
import pandas as pd
df=pd.read_csv('C:\Users\HCL\Documents\Features_Gross_1.csv')
from numpy import genfromtxt
my_data = genfromtxt('C:\Users\HCL\Documents\Features_sci_G.csv', delimiter=',')
X = my_data[:,0:19]
y = my_data[:,20]
from sklearn import preprocessing
normalized_X = preprocessing.normalize(X)
standardized_X = preprocessing.scale(X)
from sklearn import metrics
from sklearn.naive_bayes import GaussianNB
model = GaussianNB()
model.fit(X, y)
print(model)
# make predictions
expected = y
predicted = model.predict(X)
# summarize the fit of the model
print(metrics.classification_report(expected, predicted))
print(metrics.confusion_matrix(expected, predicted))
```

Decision Tree Classifier

```
import numpy as np
import io
import pandas as pd
df=pd.read_csv('C:\Users\HCL\Documents\Features_Gross_1.csv')
print df
```

```
from numpy import genfromtxt
my_data = genfromtxt('C:\Users\HCL\Documents\Features_sci_G.csv', delimiter=',')
X = my_data[:,0:19]
y = my_data[:,20]
from sklearn import preprocessing
normalized_X = preprocessing.normalize(X)
standardized_X = preprocessing.scale(X)
from sklearn import metrics
from sklearn.tree import DecisionTreeClassifier
# fit a CART model to the data
model = DecisionTreeClassifier()
model.fit(X, y)
print(model)
# make predictions
expected = y
predicted = model.predict(X)
# summarize the fit of the model
print(metrics.classification_report(expected, predicted))
print(metrics.confusion_matrix(expected, predicted))
```

Support Vector Machine

```
import numpy as np
import io
import pandas as pd
df=pd.read_csv('C:\Users\HCL\Documents\Features_Gross_1.csv')
print df
from numpy import genfromtxt
my_data = genfromtxt('C:\Users\HCL\Documents\Features_sci_G.csv', delimiter=',')
X = my_data[:,0:19]
y = my_data[:,20]
from sklearn import metrics
from sklearn.svm import SVC
# fit a SVM model to the data
model = SVC()
```

```
model.fit(X, y)
print(model)
# make predictions
expected = y
predicted = model.predict(X)
# summarize the fit of the model
print(metrics.classification_report(expected, predicted))
print(metrics.confusion_matrix(expected, predicted))
```

Precision Recall Curve

```
import matplotlib.pyplot as plt
import numpy as np
from sklearn import svm, datasets
from sklearn.metrics import precision_recall_curve
from sklearn.metrics import average_precision_score
from sklearn.cross_validation import train_test_split
from sklearn.preprocessing import label_binarize
from sklearn.multiclass import OneVsRestClassifier
# import some data to play with
iris = datasets.load_iris()
X = iris.data
y = iris.target
# Binarize the output
y = label_binarize(y, classes=[0, 1, 2])
n_classes = y.shape[1]

# Add noisy features
random_state = np.random.RandomState(0)
n_samples, n_features = X.shape
X = np.c_[X, random_state.randn(n_samples, 200 * n_features)]

# Split into training and test
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=.5,
```

```
                                    random_state=random_state)

# Run classifier
classifier = OneVsRestClassifier(svm.SVC(kernel='linear', probability=True,
                                 random_state=random_state))
y_score = classifier.fit(X_train, y_train).decision_function(X_test)

# Compute Precision-Recall and plot curve
precision = dict()
recall = dict()
average_precision = dict()
for i in range(n_classes):
    precision[i], recall[i], _ = precision_recall_curve(y_test[:, i],
                                                        y_score[:, i])
    average_precision[i] = average_precision_score(y_test[:, i], y_score[:, i])

# Compute micro-average ROC curve and ROC area
precision["micro"], recall["micro"], _ = precision_recall_curve(y_test.ravel(),
    y_score.ravel())
average_precision["micro"] = average_precision_score(y_test, y_score,
                                                     average="micro")

# Plot Precision-Recall curve
plt.clf()
plt.plot(recall[0], precision[0], label='Precision-Recall curve')
plt.xlabel('Recall')
plt.ylabel('Precision')
plt.ylim([0.0, 1.05])
plt.xlim([0.0, 1.0])
plt.title('Precision-Recall example: AUC={0:0.2f}'.format(average_precision[0]))
plt.legend(loc="lower left")
plt.show()

# Plot Precision-Recall curve for each class
```

```
plt.clf()
plt.plot(recall["micro"], precision["micro"],
        label='micro-average Precision-recall curve (area = {0:0.2f})'
              ".format(average_precision["micro"]))
for i in range(n_classes):
    plt.plot(recall[i], precision[i],
            label='Precision-recall curve of class {0} (area = {1:0.2f})'
                  ".format(i, average_precision[i]))

plt.xlim([0.0, 1.0])
plt.ylim([0.0, 1.05])
plt.xlabel('Recall')
plt.ylabel('Precision')
plt.title('Extension of Precision-Recall curve to multi-class')
plt.legend(loc="lower right")
plt.show()
```

ROC Curve

```
import numpy as np
import matplotlib.pyplot as plt
from sklearn import svm, datasets
from sklearn.metrics import roc_curve, auc
from sklearn.cross_validation import train_test_split
from sklearn.preprocessing import label_binarize
from sklearn.multiclass import OneVsRestClassifier
from scipy import interp

pd.read_csv('C:\Users\HCL\Documents\Features_sci_G.csv')
df=pd.read_csv('C:\Users\HCL\Documents\Features_sci_G.csv')
print df
from numpy import genfromtxt
my_data = genfromtxt('C:\Users\HCL\Documents\Features_sci_G.csv', delimiter=',')
X = my_data[:,0:19]
y = my_data[:,20]
```

```
y = label_binarize(y, classes=[1,2,3,4,5])
n_classes = y.shape[1]
# Add noisy features to make the problem harder
random_state = np.random.RandomState(0)
n_samples, n_features = X.shape
X = np.c_[X, random_state.randn(n_samples, 200 * n_features)]
# shuffle and split training and test sets
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=.5,
                                                    random_state=0)
# Learn to predict each class against the other
classifier = OneVsRestClassifier(svm.SVC(kernel='linear', probability=True,
                                 random_state=random_state))
y_score = classifier.fit(X_train, y_train).decision_function(X_test)
# Compute ROC curve and ROC area for each class
fpr = dict()
tpr = dict()
roc_auc = dict()
for i in range(n_classes):
    fpr[i], tpr[i], _ = roc_curve(y_test[:, i], y_score[:, i])
    roc_auc[i] = auc(fpr[i], tpr[i])
# Compute micro-average ROC curve and ROC area
fpr["micro"], tpr["micro"], _ = roc_curve(y_test.ravel(), y_score.ravel())
roc_auc["micro"] = auc(fpr["micro"], tpr["micro"])
##############################################################################
# Plot of a ROC curve for a specific class
plt.figure()
plt.plot(fpr[2], tpr[2], label='ROC curve (area = %0.2f)' % roc_auc[2])
plt.plot([0, 1], [0, 1], 'k--')
plt.xlim([0.0, 1.0])
plt.ylim([0.0, 1.05])
plt.xlabel('False Positive Rate')
plt.ylabel('True Positive Rate')
plt.title('Receiver operating characteristic example')
plt.legend(loc="lower right")
```

```
plt.show()
##############################################################################
# Plot ROC curves for the multiclass problem
# Compute macro-average ROC curve and ROC area
# First aggregate all false positive rates
all_fpr = np.unique(np.concatenate([fpr[i] for i in range(n_classes)]))
# Then interpolate all ROC curves at this points
mean_tpr = np.zeros_like(all_fpr)
for i in range(n_classes):
    mean_tpr += interp(all_fpr, fpr[i], tpr[i])
# Finally average it and compute AUC
mean_tpr /= n_classes
fpr["macro"] = all_fpr
tpr["macro"] = mean_tpr
roc_auc["macro"] = auc(fpr["macro"], tpr["macro"])
# Plot all ROC curves
plt.figure()
plt.plot(fpr["micro"], tpr["micro"],
         label='micro-average ROC curve (area = {0:0.2f})'
               ".format(roc_auc["micro"]),
         linewidth=2)
plt.plot(fpr["macro"], tpr["macro"],
         label='macro-average ROC curve (area = {0:0.2f})'
               ".format(roc_auc["macro"]),
         linewidth=2)

for i in range(n_classes):
    plt.plot(fpr[i], tpr[i], label='ROC curve of class {0} (area = {1:0.2f})'
                                   ".format(i, roc_auc[i]))
plt.plot([0, 1], [0, 1], 'k--')
plt.xlim([0.0, 1.0])
plt.ylim([0.0, 1.05])
plt.xlabel('False Positive Rate')
plt.ylabel('True Positive Rate')
```

```
plt.title('Some extension of Receiver operating characteristic to multi-class')
plt.legend(loc="lower right")
plt.show()
```

Script for Tensorflow Linear classifier

```
from numpy import genfromtxt
my_data = genfromtxt('deep_new1_1.csv', delimiter=',')
from sklearn.cross_validation import train_test_split
X = my_data[:,-1]
y = my_data[:,1]
X_train,X_test,y_train,y_test=train_test_split(X,y,test_size=0.3,random_state=0)
classifier = skflow.TensorFlowLinearClassifier(n_classes=5)
classifier.fit(X_train,y_train)
score = metrics.accuracy_score(y, classifier.predict(X))
print("Accuracy: %f" % score)
```

Script for TensorflowDeepNeuralNetworkclassifier

```
from numpy import genfromtxt
my_data = genfromtxt('deep_new1.csv', delimiter=',')
from sklearn.cross_validation import train_test_split
X = my_data[:,-1]
y = my_data[:,1]
X_train,X_test,y_train,y_test=train_test_split(X,y,test_size=0.2,random_state=0)
from sklearn import metrics
classifier = skflow.TensorFlowDNNClassifier(hidden_units=[70,80,70], n_classes=5)
classifier.fit(X_train,y_train)
score = metrics.accuracy_score(y, classifier.predict(X))
print("Accuracy: %f" % score)
```