PREDICTION OF CANCER-RELATED MUTATION IMPACT ON PROTEIN ACTIVITY USING MACHINE LEARNING

by

BRENT CANNON LIPPERT

(Under the Direction of Khaled Rasheed)

ABSTRACT

Machine Learning (ML) methods have been increasingly employed in the genetics domain. ML methods have shown promise in the field of characterizing genetic mutations. Mutations can have significant impact on the activity of the Human Epidermal Growth Factor Receptor (EGFR), a protein instrumental in cell proliferation. Over-activation of EGFR is a major cause of tumor growth. Although many computational methods have been proposed to identify disease causing mutations, these methods are not designed to predict mutation impact on protein activity. We explored feature selection strategies suitable for the small, complex data within this domain and tested a variety of machine learning algorithms. We generated a model achieving 85.9% accuracy and an F-Measure of 0.70 with a Support Vector Machine with a Gaussian radial basis function kernel using a set of 6 features. This classifier combined with others using weighted probability voting achieved an area under the ROC curve of 0.83.

INDEX WORDS: Machine Learning, Bioinformatics, Genetics, Mutation, Kinase, Protein, Cancer, EGFR, DNA, Naïve-Bayes, Random Forest, SVM, Nearest Neighbor, Logistic Regression

PREDICTION OF CANCER-RELATED MUTATION IMPACT ON PROTEIN ACTIVITY USING MACHINE LEARNING

by

BRENT CANNON LIPPERT

B.S., University of Georgia, 2017

A Thesis Submitted to the Graduate Faculty of The University of Georgia in Partial Fulfillment of the Requirements for the Degree

MASTER OF SCIENCE

ATHENS, GEORGIA

2018

© 2018
Brent Cannon Lippert
All Rights Reserved

PREDICTION OF CANCER-RELATED MUTATION IMPACT ON PROTEIN ACTIVITY USING MACHINE LEARNING

by

BRENT CANNON LIPPERT

Major Professor:	Khaled Rasheed
Committee:	Natarajan Kannan Frederick Maier

Electronic Version Approved:

Suzanne Barbour
Dean of the Graduate School
The University of Georgia
May 2018

TABLE OF CONTENTS

Page

LIST OF TABLES

Page

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

Computational methods have shown promise in combating the challenges of big data in the post-genomic era. The field of characterizing mutation effects has benefited greatly from work such as the development of tools for specifically managing and analyzing mutation data (Cario and Witte 2018), deep learning of mutational relations in scientific literature (Pan et al. 2017), and expanding online databases (Yi et al. 2017). In an attempt to harness this big data, general classifiers for point mutation-disease relatedness have become a popular topic of investigation. Early algorithms attempted to create classifiers by accounting for individual factors such as sequence conservation (Vaser et al. 2016), mutation frequency (Puente et al. 2011), and structural impact (Shi and Moult 2011). While achieving modest performance, it eventually became apparent that inferring a deleterious mutation could not be accomplished by one factor alone.

To account for more factors, a systems biology approach to predicting mutation impact became increasingly popular. Recent classifiers of deleterious mutations include FunSeq2 (Fu et al. 2014), PredSAV (Pan et al. 2017), PON-PS (Niroula et al. 2017), SusPect (Yates et al. 2014), ELAPSIC (Berliner et al. 2014), STRUM (Quan et al. 2016), and PolyPhen-2 (Adzhubei et al. 2010). This new set of classifiers commonly take into account a wide variety of features such as sequence conservation, structural properties, interaction networks, and gene ontology annotations. The classifiers implement algorithms such as support vector machine (Yates et al. 2014), Bayesian methods (Adzhubei et al. 2010), Random Forest (Niroula and Vihinen 2017)(Pons et al. 2016), and a variety of ensemble methods that implement either weighted or unweighted voting from multiple classifiers (Quan et al. 2016)(Berliner et al. 2014)(U et al. 2014)(Pan et al. 2017). However, the dimensionality of the dataset may be too large. Furthermore, the concept of whether a mutation is deleterious may be too abstract and broad for the current classifiers. Classifiers trained to predict mutation-disease relationship tend to over-fit the training set and may not perform as well when classifying new data (Gnad et al. 2013).

Narrowing the scope of prediction, some classifiers focus on subsets of mutations such as those in kinases which are frequently observed in cancer. Recent work includes KinMutRF (Pons et al. 2016) predicting deleterious kinase mutations and predicting cancer driver mutations (U et al. 2014). Feature selection in both studies revealed that kinase-specific features are potentially more important than generalizable features. This observation suggests an advantage in using specialized classifiers that take into account kinase-specific features which are missed in genome-wide datasets. This allows the classifier to be more generalizable to unseen data (U et al. 2014).

Computational methods have also been applied in predicting concrete measures such as biochemical features. Recent works in mutational impact have produced statistical and machine learning models that predict change in protein folding free energy ($\Delta\Delta G$) (Giollo et al. 2014)(Quan et al. 2016), changes in free energy of binding ($\Delta\Delta G_{binding}$) (Dehouck et al. 2013), and thermal stability (ΔT_m) (Pucci et al. 2016). A predictor for mutation impact on kinase-substrate phosphorylation has also been developed (Wagih et al. 2015). Many predictors of biochemical property usually involve fitting a customized statistical model onto concrete experimental data. The predicted values may, in turn, be used to infer potential deleterious effects in the context of larger biological systems.

Avoiding the pitfalls in predicting disease-related mutations, we recently established a machine learning classifier for identifying kinase-activating mutations in human EGFR. These are commonly observed in lung cancers amongst other phenotypes. As a member of the tyrosine kinase family, EGFR is frequently mutated in many different cancer patients making it a useful target of study. As a side note, there has recently been a high throughput experimental method of characterizing the effects of EGFR mutations on cell growth (Kohsaka et al. 2017).

In this study, we seek to develop classifiers for mutations on EGFR-ligand independent activity. We manually curated a list of 77 distinct mutations in the EGFR kinase domain from published literature. Each mutation was classified as either “activating” denoting increased ligand-independent (EGF-independent) phosphorylation activity relative to the wild type (WT) or “non-activating” denoting either similar or decreased activity relative to the wild type. The acquisition of activating EGFR mutations often results in aberrant cell growth. We developed a novel and extensive feature set that includes the physicochemical properties of residues, various empirical energy functions, kinase-specific evolutionarily conserved residues, EGFR-specific assembly interface, and kinase conformation-specific structural properties. Iterative feature selection identified six most informative features in predicting activating mutations: (1) the difference in Rosetta energy (O’Meara et al. 2015) between the wild type and mutant protein whilst in the inactive of EGFR (diff_relax_inactive), (2) the difference in accessible surface area between the wild type and mutant protein whilst in the active state of EGFR (diff_asa_active), (3) the entropy of the mutation site when aligned to all members of the tyrosine kinase family (conservation_tk), (4) whether the mutation localizes in the EGFR asymmetric dimer interface (active_dimer), (5) the entropy of the mutation site when aligned to all EGFR kinases (conservation_egfr), and (6) the difference in the residue distance network closeness-centrality as calculated by NetworkX (Hagberg et al. 2008) between the wild type and mutant residue in the active conformation of EGFR (diff_active_network_closeness). The support vector machine yields an accuracy of 82.7% with an F-Measure of 0.615 based on the 10-fold cross-validation and is the most effective model from our study.

CHAPTER 2

MATERIALS AND METHODS

Data source

The ligand-independent autophosphorylation activity of EGFR is typically measured by cell-based western blot assay. By mining the existing literature, we manually quantified the relative activity level of 77 different EGFR point mutations in the kinase domain using digital densitometry (ImageJ) from 8 different published studies (Zhang et al. 2006)(Jura et al. 2009)(U et al. 2014)(Ruan et al. 2017)(Chen et al. 2005)(Choi et al. 2006)(Kancha et al. 2009)(Mcskimming et al. 2015). The ratio between tyrosine phosphorylation and the total expression level of EGFR is taken as the numerical activity score for a given mutation. In order to form two discrete classes a threshold of 1.5 was chosen after which any mutation with a score below that threshold was labeled as Non-Activating and those with scores greater than or equal to it were labeled as Activating. The list of EGFR mutations and corresponding references can be found in the S1 Table in Supporting Information.

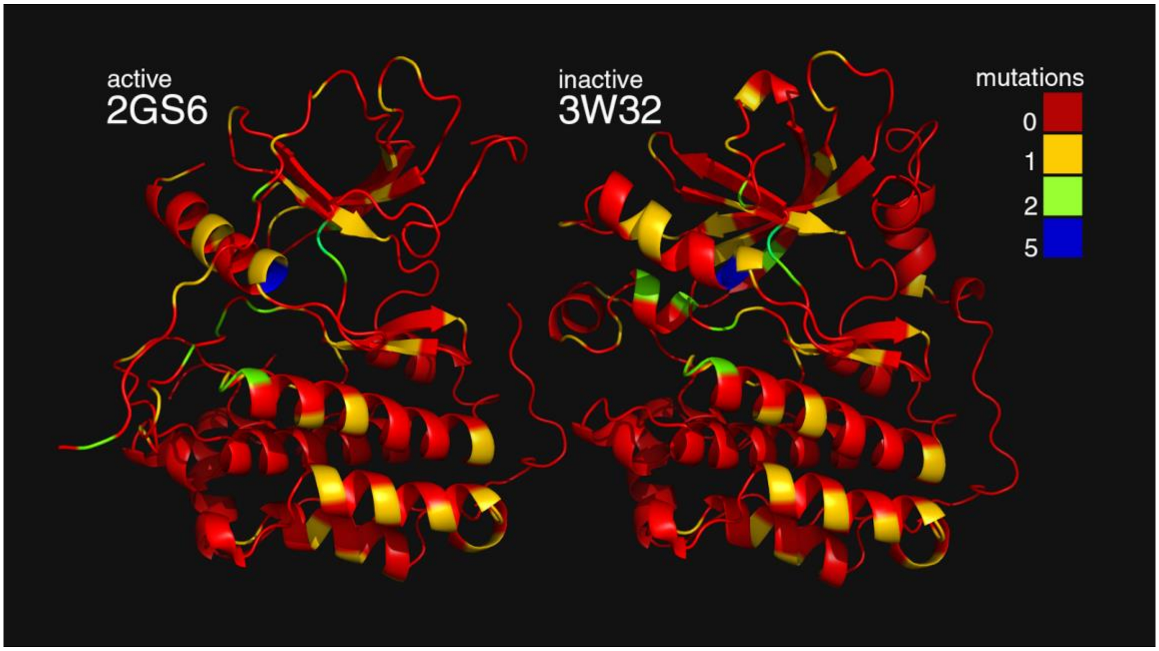

Figure 1. Mutations of Protein Structure: Locations and counts of point mutations within the dataset shown on the reference PDBs for the active and inactive conformation.

Training Features

Our training features can be broadly classified into several categories which will be described in Table 1 below. The label “2 conformations” denotes that the feature is calculated for the both active and inactive conformation.(Velankar 2013)

Table 1. Descriptions of Categories of Training Features

Structure	Difference in b-factor compared to WT (2 conformations) Difference in surface area compared to WT (2 conformations) Whether the mutation localizes in the EGFR dimer interface
Biochemical	Difference in polarity, hydropathy, volume, charge, and molecular weight compared to WT.
Conservation	The mutation’s position-wise entropy when aligned to: all eukaryotic protein kinases, tyrosine kinases, EGFR family kinases, and EGFR kinases.
Energy	Difference in Rosetta energy compared to WT (2 conformations) Difference in Rosetta pmut scan score compared to WT (2 conformations)

Structural Network	Difference in network betweenness compared to WT (2 conformations) Difference in network closeness compared to WT (2 conformations)
Others	Frequency of the point mutation in the COSMIC database

Preprocessing

After the data collect described above, we had curated a set of 77 mutations. The resulting Activating class contained 21 mutations whereas the Non-Activating contained 56, making it 2.67 times the size of the Activating class. Due to this imbalance between the Activating and Non-Activating class, class-weighting was used in all feature section and model training (Gustavo et al. 2004). A set of 35 features was then defined and calculated for each mutation (see Table 1.). Some of the features within our dataset had a much larger range of values than other features and as such could potentially bias the model to favor them over the smaller value features. In order to solve this problem, we used standardization, a data scaling technique that transforms each feature to have a mean of 0 and unit-variance (Shanker et al. 1996).

There were a few cases in which certain features were not able to be calculated for certain mutations however this was uncommon and the resulting dataset was ~96% dense.(Wilson et. Al 2003) In order to address the missing values, several approaches were considered. Due to the limited size of our dataset, we decided against discarding those mutations which had missing values. Therefore, a method of imputation had to be chosen and employed. We experimented with several approaches that calculate artificial values for those missing by using the values that are available. These include techniques such as Mean Substitution and Expectation Maximization.(Gold and Bentler 2000) The issue with using these techniques is the small size of our dataset compared to its high dimensionality and feature complexity hinders these imputation algorithms from effectively calculating meaningful artificial feature values, causing them to instead inject noise into the data. In the end, we found that the imputation technique that maximized the performance of our models was simply filling in all missing values with 0's.

Feature selection

Ideally feature selection would be performed on a separate set entirely than the training and testing sets, (Blum and Langley 1997) due to the extremely limited size of our dataset we had no choice but to perform feature selection on the same set we would later classify on. Additionally, our dataset has a very high degree of dimensionality relative to its size. As such, great care had to be employed to avoid data leakage or over-fitting of our model. (Ye and Wang 2006) Data leakage is when a model receives extra information for training than what is contained in the training data which may bias its performance and allow it to perform unrealistically well. In our case, performing feature selection on the same set as model training/validation allows the best features for validation to leak into selection, biasing the model. Over-fitting is when a model learns a training set overly-well, making it perform highly on that specific data but poorly on data it did not train on. (Ye and Wang 2006) In our data, the high degrees of freedom potentially allowed to our models could easily over-fit such a small dataset. Feature selection had to be performed in such a manner such that dimensionality reduction could be performed on our limited data whilst minimizing the potential for overfitting and data leakage during training/validation.(Arlot and Celisse 2010)

We determined that the most effective approach would be to integrate feature selection into our experiments' 10-fold cross-validation. 10-fold cross-validation is a method which pseudo-randomly partitions the data into 10 sets of roughly equal size, wherein each set it then used as a validation set for the remainder of the data. This prevents the leakage of validation data into the training set. (Martens and Dardenne 1998)(Arlot and Celisse 2010) Then within each fold, our feature selection technique was applied. The advantage of this approach is features are never chosen from the same data that will be used to validate the model, preventing any data leakage. A disadvantage however is due to feature selection being performed for each fold it will be run 10 times per experiment, drastically increasing the time taken for experimentation. Additionally, it is possible each fold may choose a completely distinct feature set from the others, making post-experiment analysis of each features' predictive power more difficult. Despite these limitations we found this approach to be highly effective in combating the issues that arise from small but complex data such as ours.

The feature selection technique used within each fold is described as follows. Sampling with replacement was done to produce a sample set *s* that is 75% the size of the original. Then Correlation-based Attribute Evaluation (Hall 1999) was used as a feature selection metric. This method determines the merit of feature subsets for classification by measuring their member features correlation to each class as well as lack of correlation to each other. The merit for each feature set is defined by the following formula (Hall 1999):

$$M_s = \frac{k\overline{r_{cf}}}{\sqrt{k+k(k-1)\overline{r_{ff}}}} \qquad (1)$$

Where M is the merit of subset s with k features, $\overline{r_{cf}}$ is the mean correlation of the features to the classes and $\overline{r_{ff}}$ is the mean of the features' correlation to other features. This method was used in conjunction with Best First Search. Best First Search starts with an empty set of features then adds one feature at a time, evaluating each resulting subset and chooses the best of these neighbor subsets to be used in its next iteration. The algorithm backtracks should it ever reach a point in which the current neighbor subsets it is checking perform worse than some previous subset. Should the algorithm ever explore some predefined number, in our case 10000, of consecutive non-improving subsets it will terminate and return the best subset it has found thus far (there is no specific significance in the value of 10000 it is simply how much you want to allow the algorithm to continue without termination, without this cut-off the search would explore every single possible subset, in our case 2^{35}). (Patankar and Chavda 2016).

Then, the Best First Search was repeated using Wrapper Subset Evaluation for the search heuristic. This is a technique in which each feature subset is used for training some classifier with 10-fold cross-validation then using an evaluation metric, in our case the F-measure with respect to the Activating class, as the subset's merit. (Arlot and Celisse 2010) This was repeated with all of our models (see Classification Methods for details on the models used)

This process was repeated over one hundred unique 75% samples. Aggregate scores for how the individual features performed across the hundred samples were then calculated. Voting was used in which each feature in the selected subset for a sample gets one vote; the votes are totaled across all samples generating a ranking for each individual feature. The feature rankings were then analyzed to find the top set of high-ranking features with the largest gap to the next highest feature below that set. The resulting set was then used as the feature set for training the models within that fold. The feature selection process was repeated for each fold before training occurred.

The advantage of this process is it incredibly rigorous; testing a vast number feature sets across a variety of subsamples, using two different feature set evaluation metrics, and employing all of the models to be tested in the wrapper results in a feature set with a high probability being predictive and generalizable. The obvious disadvantage is a single experiment take hours to execute on our dataset of just 77 data points, making it only practical to use in Small Data applications (though with more data less rigor is required regardless).

Classification Methods

Sci-Kit Learn was the library used for our study.(Pedregosa *et al.*, 2011) The experiments were repeated 100 times to obtain the final results. Class weighting was employed in all training in order to account for the imbalance.(Hall 1999) We experimented with a large variety of models, a selection of which we shall describe..

The Random Forest classifier works by developing an ensemble of decision trees. (Breiman 2001) Decision trees are a classification method in which a tree structure if formed such that at each interior node a single rule is defined which evaluates an attribute of some input data point and determines which of the node's children the data point will be evaluated by next. This process is repeated until the data point reaches a leaf node which then determines the data point's class. A random forest model builds a series of these trees, first by selecting a pseudo-random subset of features of a predefined size from the original input feature set. Then, a sample of the training data is chosen using sampling with replacement. A decision tree is then built on this sample using the feature subset. Any decision tree construction algorithm may be employed, in our case it was C4.5 (Ruggieri 2002). A series of trees are built in this manner. After the model is built, to classify each point it is classified by each tree in the model then the class which had the most trees select it is chosen as the overall classification. By training an ensemble of trees each on separate subsets and each using a random subset of features Random Forest classifiers are very effective at combating over-fitting. (Breiman 2001)

Nearest Neighbor based models classify unseen data points based on their proximity to already labeled points in the feature space. No training is actually required for the model. Each target data point is compared to the other data points in the dataset. Those nearest to the data point are selected to be used in classification. The number of points used can be decided in manners such some predefined fixed value, as in K-Nearest Neighbors (Denoeux 1995), or the points found within some fixed radius as in Radius Nearest Neighbors. (Cover 1967) After the nearest neighbors are selected, each neighbor votes on the classification for the unlabeled point and the class with the most votes is assigned to the unlabeled point. If the classes are unbalanced, the classes are weighted giving the smaller classes more influential votes. Ties are broken by adding the next nearest neighbor to the set and considering its vote.

A Support Vector Machine is a classifier designed to find one or more dividing hyperplanes between separable classes within a complex, multi-dimensional feature space. It works by finding one or more hyperplanes in the feature space that divides the space into two separate sub-spaces, each correlated to one of two classes of data. Unseen data points are then classified by finding which sub-space they fall into. The hyperplanes considered the best are that with the maximum distance to two support vectors on opposite sides of the plane. A support vector is a vector representing the edge of one of the separable classes in the training space. (Scholkopf et al. 1997) The hyperplane is calculated using a kernel. The kernel/resulting hyperplane need not be linear. The common kernels used for support vector machines include linear, polynomial, hyperbolic tangent, and Gaussian radial basis functions. (Scholkopf et al. 1997)

Logistic Regression is a classifier that calculates the probability of a feature vector belonging to a particular class. (Dreiseitl 2002) It works by training a linear function in the form of the summation of each value in the feature vector after each multiplied by some corresponding weight. It then transforms the result of this equation using the logistic function with a maximum of 1, giving a probability value for the given point belonging to some class. This probability is then divided by the probability of the data point not belonging to that glass, resulting in a comparative likelihood value for the data point belonging to the given class versus the other. The linear function used in the probability function is trained using any technique that can be used for training a standard linear regression model (in our case stochastic gradient descent). (Dreiseitl 2002)

A Naïve-Bayes classifier is one based on the application of Bayes theorem. (Lewis 1998) Bayes Theorem estimates the probability for a given event based on prior knowledge of conditions that may be correlated. A Bayesian classifier uses the feature values of the training data as the prior conditions to calculate the probability. The Naïve-Bayes is known as naïve because is it makes the assumption that the features are all entirely independent of each other when making its probability calculations. While this can somewhat hinder the classifier, as in many cases the features are not truly independent, these classifiers still perform surprisingly well despite their simplicity (Lewis 1998). Gaussian Naïve-Bayes is a formulation of the Naïve-Bayes classifier that works with continuous feature values.(Lou et al. 2014) In this formulation, the features are used to calculate some Gaussian distribution modeling that feature’s possible values to a probability for each class.

CHAPTER 3

RESULTS

Results and Discussion

As discussed above (see Feature Selection), for every 10-fold cross-validation of the models it was possible for up to 10 distinct feature sets to be chosen. We ran 100 unique 10-fold cross-validations with a different seed used in the pseudo-random number generator for every fold to obtain the results for an experiment. Therefore, for each experiment up to 1000 unique feature could have been generated. With this in mind, a significant result of our study was that in the experiment which employed all of the models described above, a singular feature set of size 3 was chosen across all 1000 chosen sets. It was a set of three features, diff_relax_inactive, diff_asa_active, and active_dimer. These three features clearly have a significant predictive power within our dataset. Due to the measures taken to prevent data leakage and over-fitting in our selection technique as well as the repeated experimentation across randomized dataset partitions, it is highly probable that these features are generalizable and will still be predictive for mutations not within our data set. Additionally, this demonstrates the robustness of our approach to feature selection, as applied to Small Data.

Table 2. Feature Scores: Average voting scores for each feature during feature selection across all experiments. Assuming the scores were similar for each fold in each experiment, it is easy to see why those three features were selected consistently with such a large gap between them and the next feature.

Feature	Score (%)
diff_relax_inactive	92.53
active_dimer	89.1
diff_asa_active	88.3
conservation_tk	49.4
conservation_egfr	43.7
diff_pmut_scan_per_residue_active	32.9
conservation_epk	29.3
conservation_egfrfam	26.7
diff_pmut_scan_active	25.2
diff_relax_per_residue_inactive	23.3
diff_active_network_closeness	23.1
blosum62	19.7
diff_polarity	17.6
diff_local_inactive_neg_tk	11.6
diff_hydropathy	10.6
diff_charge	9.8
diff_pmut_scan_per_residue_inactive	8.7
diff_relax_per_residue_active	8.6
diff_relax_active	7.9
diff_pmut_scan_inactive	3.2
diff_local_inactive_neg_egfrfam	2.8
diff_local_inactive_neg_egfr	1.6
diff_local_inactive_neg_epk	1.3

The Rosetta inactive energy is a measure of the overall fold of the protein. The kinase domain of EGFR exists in an equilibrium between the active and inactive states (Jura et al. 2009). Mutations in the kinase domain that destabilize the inactive state is likely to shift the equilibrium towards the active state, thus activate the enzyme (Zhang et al. 2006)(Ruan and Kannan 2015). The diff_relax_inactive feature measures the relative stability of the mutant in comparison to WT in the inactive state. Identification of this feature shows that modulating the stability of inactive

EGFR is a common mechanism by which cancer cells alter kinase activity. The importance of taking into account structural conformation is shown by diff_relax_inactive (92.53%) being the most important feature while diff_relax_active (7.9%) holds almost no predictive power.

The activation of EGFR requires the formation of an asymmetric dimer (Zhang et al. 2006). Mutations disrupt the interface typically result in kinase inactivation (Lavoie et al. 2014)(Zhang et al. 2006)(Ruan et al. 2016). In particular, many of our training data are mutations in the asymmetric dimer interface and inactivate the kinase by disrupting the dimer formation. Therefore, the feature active_dimer which is good at classifying a subset of mutations is selected.

The diff_asa_active describes the difference of solvent accessible area between WT and mutant EGFR. It is unclear why solvent accessible area is important for the classification. Because only the active conformation allows substrate binding, a change in surface area may also reflect a change in substrate accessibility. The importance of conformation is shown yet again as diff_asa_inactive has virtually no predictive power.

Another result of note is our feature selection doesn't choose the frequency of a mutation as important to determine the Activating mutations. This result emphasized the fact that many of the rare occurring mutations could also contribute the kinase activation. However, a systematic understanding of these mutations is currently lacking.

In our experimentation, we are most interested in the ability to classify the activating mutations. Our focused interest in the activating class along with the severe imbalance in our data set, which contains 56 Non-activating mutations but only 21 Activating mutations, so in our evaluation of our models we judged them primarily on their prediction ability of the activating class. The metrics used for evaluating the models were accuracy, precision, and recall. Accuracy is the measure of how many mutations were classified correctly by the model. Recall is the measure of how many mutations that are members of certain class were correctly classified as that class. Precision is the measure of how many mutations classified as a certain class were indeed members of that class. F-Measure is a metric which combines precision and recall into a single metric that is intended to balance their relative importance. (Sokolova et al. 20006)

$$\text{Accuracy} = \frac{\#\ Correctly\ Classified}{Total\ \#\ of\ Datapoints} \qquad (2)$$

$$\text{Recall} = \frac{True\ Positives}{True\ Positives + False\ Negatives} \qquad (3)$$

$$\text{Precision} = \frac{True\ Positives}{True\ Positives + False\ Positives} \qquad (4)$$

$$\text{F-Measure} = 2 \cdot \frac{Precision \cdot Recall}{Precision + Recall} \qquad (5)$$

The three highest performing models according to these metrics were Random Forest, Naive-Bayes, and Support Vector Machine with a Gaussian Radial Basis Function kernel. These three's performance are as follows:

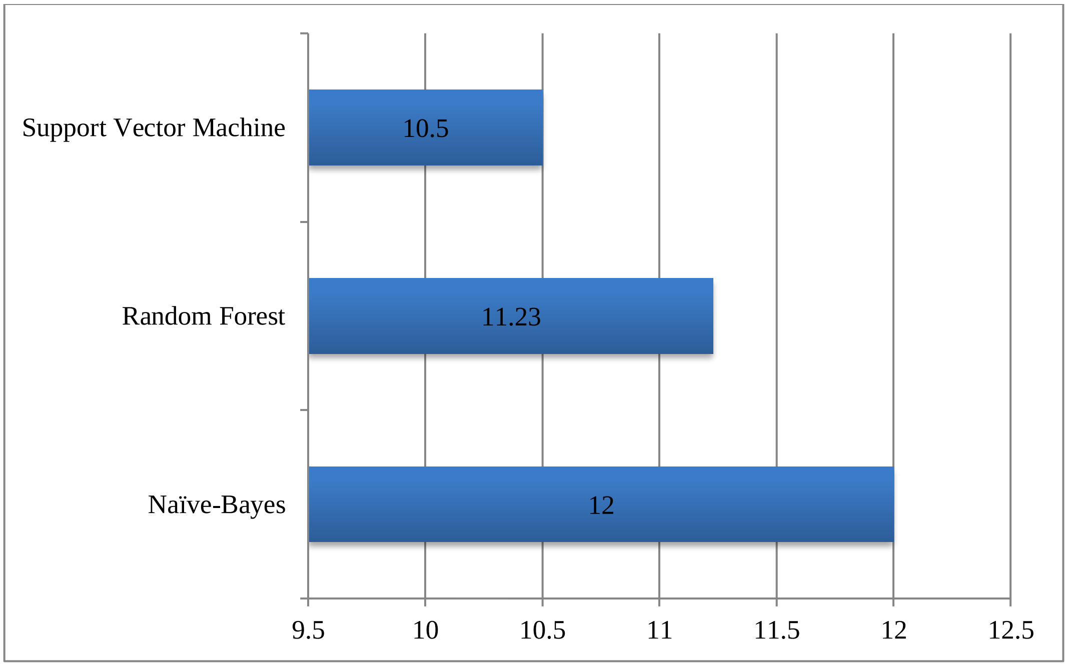

Figure 2. Average Correctly Classified Activating Mutations: The average correctly classified Activating mutations for each model across the 100 experiments. (The minimum possible value for this is 0 and the maximum 21 but for readability's sake the x-axis range has been restricted to 9.5 to 12.5)

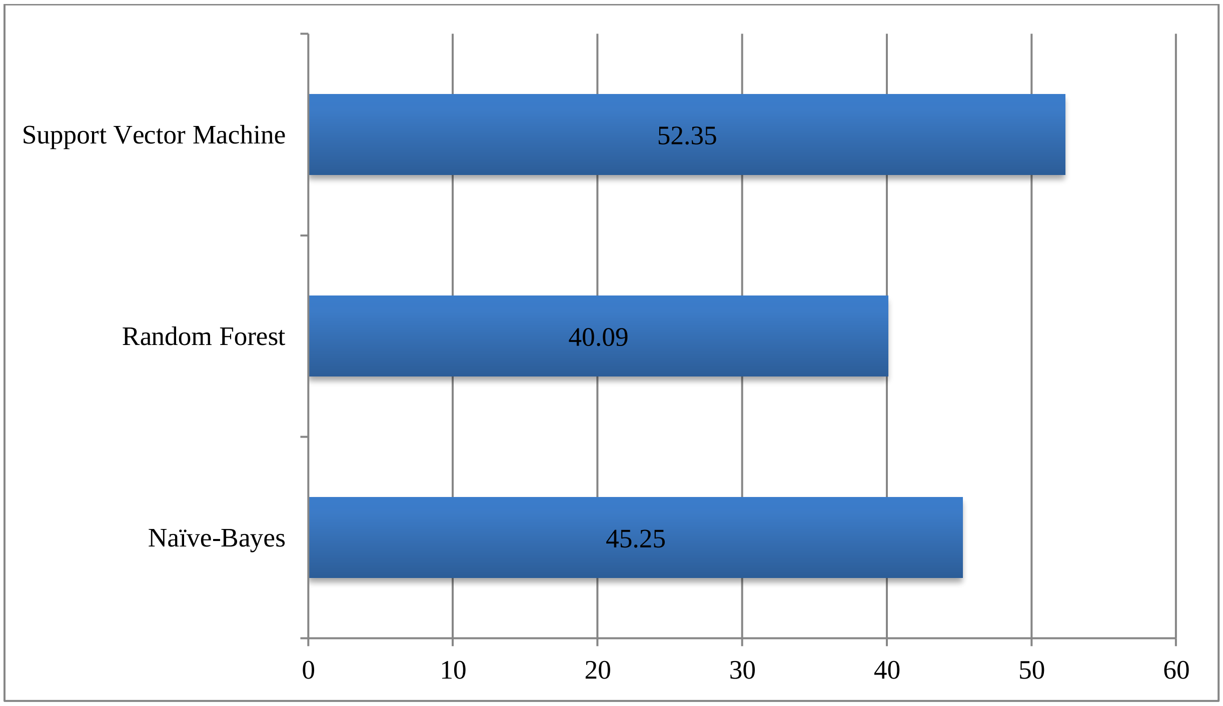

Figure 3. Average Correctly Classified Non-Activating Mutations: The average correctly classified Non-activating mutations for each model across the 100 experiments

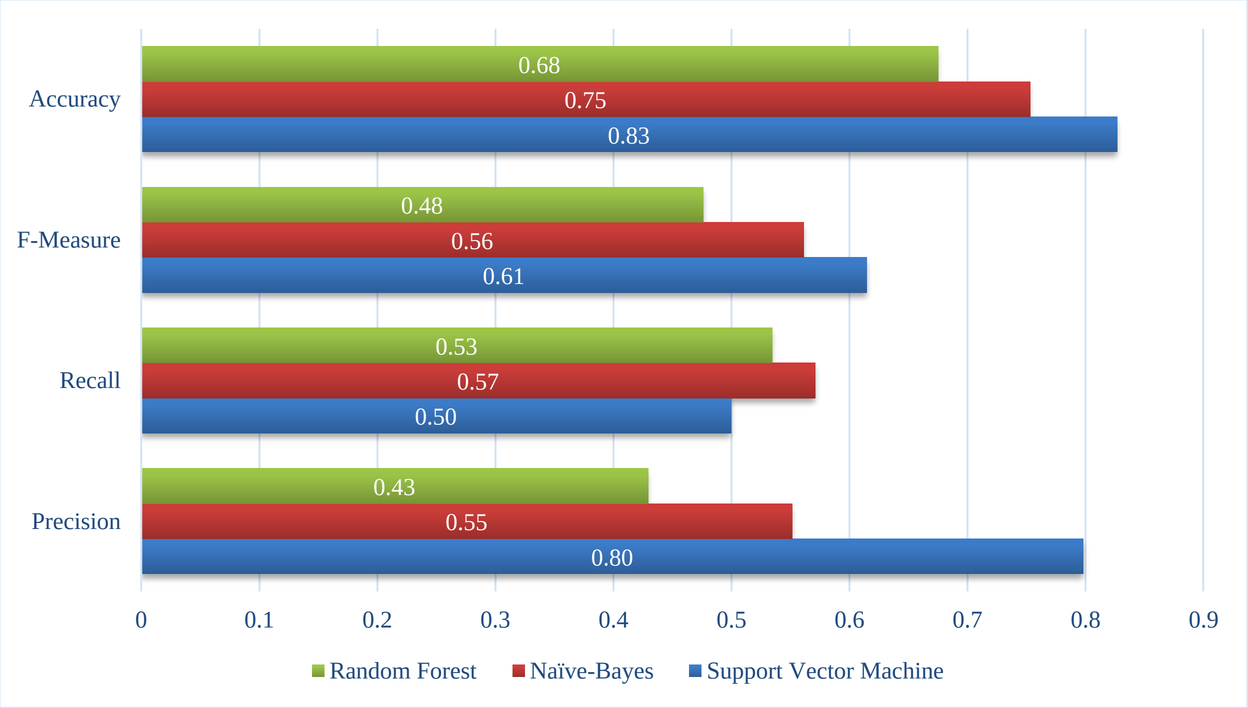

Figure 4. Classifier Performance Comparison. The average performance metrics for the classifiers across the 100 experiments. The Activating class was considered the positive class in the calculation of Recall and Precision.

Table 3. Activating Mutation Correct Predictions: The number of correct classifications for each variant within the Activating class by each model and the total across all three classifiers. (The correct classifications for Non-Activating class is in S1 Table in Supporting Information)

Variant	RF	NB	SVM	Total
M766S	100	100	100	300
M766T	99	100	100	299
D761N	96	100	100	296
I759A	92	100	100	292
L858R	96	100	94	290
T725M	94	100	94	288
E746K	31	100	80	211
G735S	87	100	0	187
Q791E	74	15	97	186
A767T	82	0	97	179
T790M	69	99	0	168

L833V	67	0	97	164
L861Q	17	0	93	110
G724S	2	100	0	102
L777F	0	100	0	100
G719S	86	0	0	86
S768I	63	0	0	63
L861R	40	0	0	40
R776C	4	28	0	32
R776H	1	0	0	1
M766A	0	0	0	0

Naïve-Bayes performed the best in sheer number of predicting the Activating class, with an average of 12 Activating mutations predicted correctly. Random Forest was second with an average of 11.23 and the Support Vector Machine was last with an average of 10.5 (Fig 2). As for the Non-activating class, in this the Support Vector Machine performed the best with an average of 52.35 true negatives predicted correctly. Naïve-Bayes was second with 45.25 and Random Forest last with 40.09 (Fig 3). In Accuracy, the Support Vector Machine was much higher than the others with an accuracy of 83%, Naïve-Bayes second with 75%, and Random Forest was last with 68% (Fig 4). The Support Vector Machine may have performed better in terms of the Non-activating class and Accuracy but as discussed above these are not the best metrics for evaluating our models; their performance with the Activating class yields more relevant metrics. Naïve-Bayes predicted the most true-positives on average and as such had the highest recall measure for the Activating class at .57, however its precision was only .55, much lower than the Support Vector Machine so its high recall may possibly be attributed to a bias towards the prediction of the Activating class in general (Fig 4). The Support Vector Machine had by far the highest Activating precision at 0.80, 0.25 higher than Naïve-Bayes in second (Fig 4). Though it did not have the highest recall, this high precision gives the Support Vector Machine the highest F-Measure, or F1 Score. Since at .61 the F1 Score for the Support Vector Machine was the highest for the Activating class (Fig 4) we can conclude that it is our highest performing model (Fig 4) and had the added benefit of performing the best on the Non-activating class as well (Figs 3).

These three models performed the highest most likely due to the limited size of our dataset. Random Forest through its technique of subsampling the data and dataset to build an ensemble is very resistant to over-fitting, something which can be a serious issue in a dataset this small. Additionally, having a series of models each trained on different features and mutation samples allows for a more thorough consideration this complex feature space, and by examining multiple random relations between the mutations and their features the Random Forest model can more accurately and in a more generalizable fashion capture the correlations between the features and each class. Naive-Bayes resists over-fitting in a completely different manner. Due to the simplicity of the Naive-Bayes classifier, it is more easily able to learn from small data sets. It has no need for large numbers of training example to calculate its event probabilities. This can give it an advantage in small datasets over algorithms such as Logistic Regression using stochastic gradient descent, which requires large number of training points it can use to iterative update its function weights in order to converge. (Lewis 1998) Finally, the highest performing model, the Support Vector Machine with a Gaussian radial basis function kernel. Support Vector Machines like Naive-Bayes do not necessarily need a large volume of data points to train on. As long as the points within the data are representative of the general areas within the feature space points of that same class will be found, a support vector machine will perform well. This is because its method of finding dividing hyper planes that maximize their distance to any support vector leaves a large margin in which unseen data points may fall outside the support vector(s) binding their class yet still be on the correct side of the classifying hyperplane(s) support vector. The Gaussian radial basis function kernel performs better than the other SVM kernels due to its increased flexibility in adapting to the support vectors (Scholkopf 1997).

After our experimentation with a multitude of classifiers found the Support Vector Machine with a Gaussian radial basis functional kernel to be the best performing, we repeated our full experiment, now only using the SVM with the optimal kernel for feature selection, leaving out the less effective classifiers. This would always yield one of 4 feature sets. These feature sets we remarkably for several reasons. Firstly, all four of these sets were all a superset of the previously found feature set. This further demonstrates the importance and predictive power of the features diff_asa_active, diff_relax_inactive, and active_dimer within our dataset. Secondly, there were two other features also consistent across all four sets. These were conservation_tk and conservation_egfr. Finally, the 6th feature was interchangeably one of the 4 Structural Network features; these being diff_active_network_closeness, diff_active_network_betweenness, diff_inactive_network_closeness, and diff_inactive_network_betweenness. To explore this phenomenon, we trained and validated the Support Vector Machine on all 4 feature sets and compared the results. All of the metrics for all 4 feature sets were essentially identical (there was some very slight variation with a magnitude on the order of a thousandth). Furthermore, when introducing more than one of these 4 features into the set the performance decreased. This indicates that in terms of predictive power all 4 Structural Network features are equivalent and using all 4 of them is redundant. When examining the 4 features’ values charted across the mutations we can see the clear correlation:

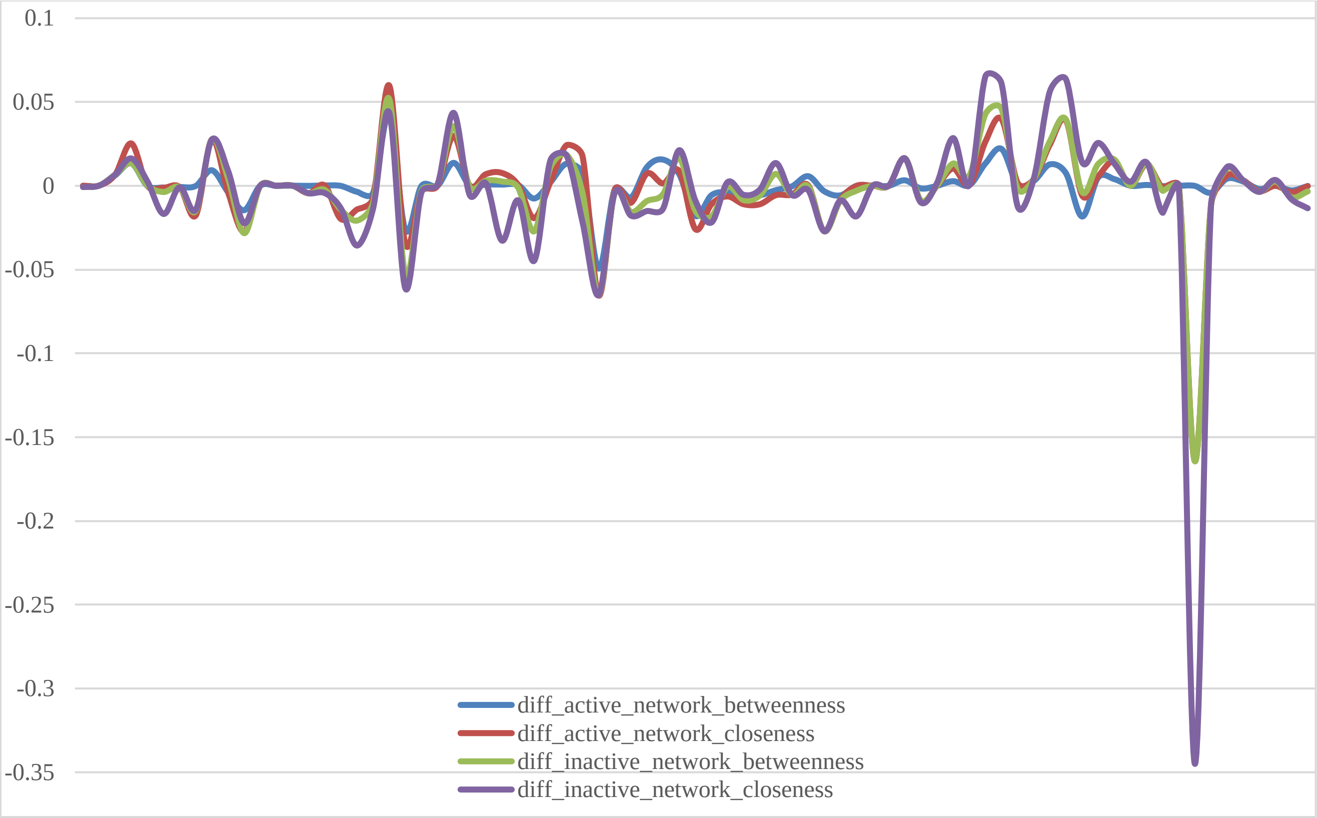

Figure 5. Network Features Across All Mutations: When charting the feature values for each mutation across the x-axis we can see a clear correlation between their values in most instances.

Thereafter we eliminated diff_active_network_betweenness, diff_inactive_network_closeness, and diff_inactive_network_betweenness from our dataset, leaving only diff_active_network_closeness (this feature was chosen arbitrarily, since the 4 are redundant any of the others could have been kept instead). The experiment was then run again. Doing so yielded the following feature rankings:

Table 4. SVM Only Feature Scores: Average voting scores for each feature during feature selection across all experiments with only the SVM with the RBF kernel being used. Again, there is a very clear distinction between the consistently selected features and the rest.

Feature	**Score (%)**
diff_relax_inactive	93.5
diff_asa_active	92.7
conservation_tk	87.4
active_dimer	86.4
conservation_egfr	84.7

diff_active_network_closeness	83.4
diff_pmut_scan_per_residue_active	25.7
conservation_epk	23.9
conservation_egfrfam	23.3
diff_pmut_scan_active	23.1
diff_relax_per_residue_inactive	21.4
blosum62	14.2
diff_polarity	14
diff_charge	10.1
diff_local_inactive_neg_tk	9.5
diff_relax_per_residue_active	7.7
diff_relax_active	5.7
diff_pmut_scan_per_residue_inactive	5.1
diff_pmut_scan_inactive	4.4
diff_hydropathy	1.1
diff_local_inactive_neg_egfr	1
diff_local_inactive_neg_egfrfam	0.8
diff_local_inactive_neg_epk	0.3

A new set of now six features was selected consistently, again chosen in nearly every experiment. It contained the same three features as selected by all of the classifiers together, but added three new features unique to the SVM. Sequence conservation features were found to be of increased importance for the SVM as compared to the other classifiers such as Naive Bayes and Random Forest (no evidence here, because in Table 1 it is not a result of using Naive Bayes or Random Forest). {The conservation features incorporated in our study are evaluated at different levels. Specifically, we calculated the conservation score (entropy) of the sequence alignment of 1} all eukaryotic protein kinases (ePKs), 2} all tyrosine kinases (TKs), 3} all EGFR family kinases, and 4} all EGFR homologous sequences. Interestingly, the 2nd level (conservation_tk) and 4th level (conservation_egfr) were found to be the most important conservation features, while the 3rd level (conservation_egfrfam) lying in between was found to be the worst. Despite numerous the popularity of using sequence conservation in predicting mutation impact (Gnad 2013), the prediction performance seems to be highly dependent on the way the sequence alignment is built. In addition, the diff_active_network_closeness is also selected as an important feature for SVM. The network properties of protein structures have previously explored to understand the allosteric communications between different regions of the protein (James and Verkhivker 2014). Our result further suggests that it might be useful to understand mutation impact as well.

In order to further explore the impact of the features we have proposed, we took the set of 6 features and did a cross-comparison of this full set versus the subsets formed by removing each category of features we proposed. E.g. we experimented with the removal of the Structural Network feature diff_active_network_closeness, the Energy feature diff_relax_inactive, and the Structural features active_dimer and diff_asa_active. In doing so we aimed to demonstrate the importance of EGFR-specific features in the classification of these mutations. Our comparison was done by generating the receiver operating characteristic (ROC) curve for each set. The ROC curve charts the true positive rate of a classifier against its false positive rate.(Hanley and Mcneil 1982) This is done by having the trained classifier predict the probability of each instance in the validation set belonging to the Activating class. The samples with a probability higher than some threshold are predicted as Activating. This threshold starts at 1 (meaning nothing is classified as Activating) and is then iteratively decreased, with the false positive and true positive rate for the classifier at each new threshold being calculated. As the threshold becomes less stringent, both the false and true positive rate will increase. The ROC curve charts how the true positive rate changes as compared to the false positive rate. These curves can be compared by calculating their Area Under the Curve (AUC) .(Hanley and Mcneil 1982) The higher the AUC, the more likely the classifier will correctly predict the Activating class. The classifier used for the ROC curve was the Support Vector Machine.

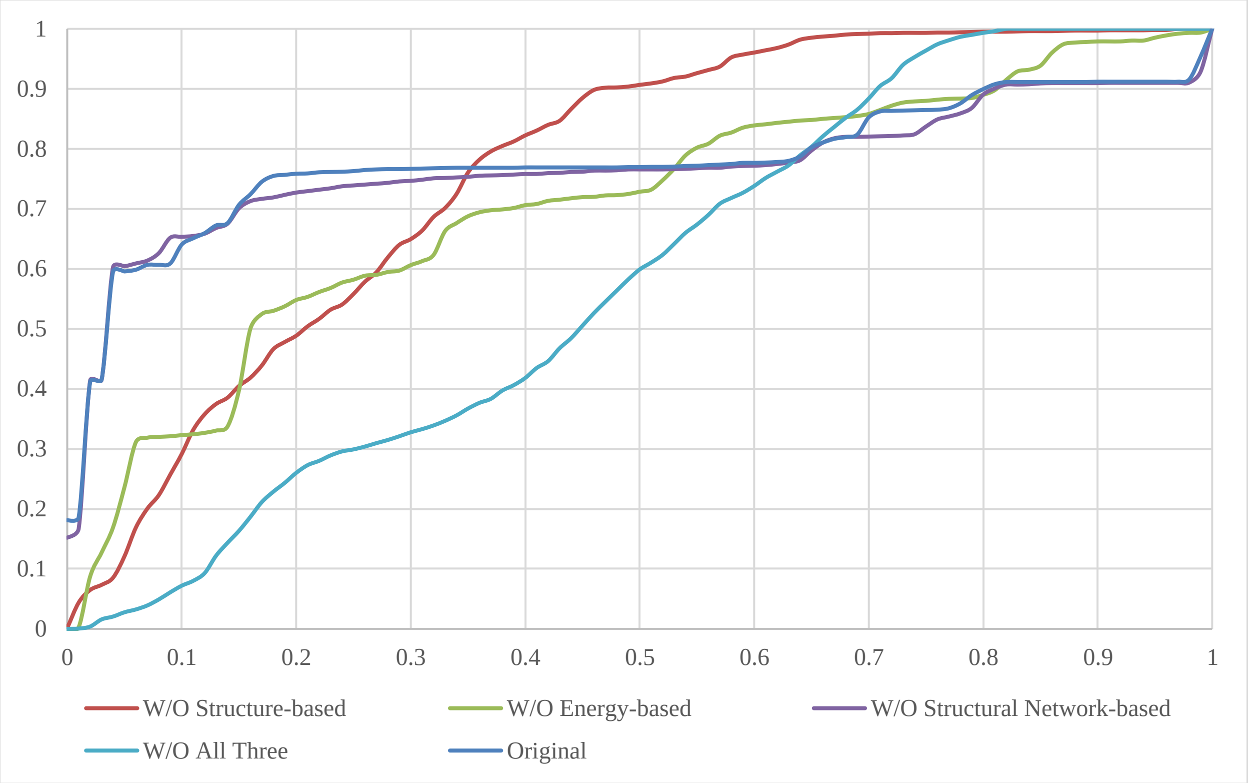

Figure 6. Feature Sets' ROC Curves: Compares the full set of 6 features to the removal of each category of features. False Positive Rate is the x-axis and True Positive Rate the y-axis. Generated using an SVM for training/validation

Table 5. AUC for Figure 5.

Feature Set	AUC
Original	0.759211429
W/O Structure-based	0.732628788
W/O Energy-based	0.663242944
W/O Network-based	0.749817446
W/O All Three	0.490396147

As demonstrated by Figure 6, the removal of the Structure-based features or the Energy-based features both significantly diminish the model's performance in terms of its ability to predict the Activating class while minimizing false positives. Although, interestingly the set excluding the structural features did perform better at higher false positive rates than the original set. This accounts for why its AUC is only ~0.03 less than the original set. The removal of the

Structural Network feature had little effect on the classifier's performance, only reducing the AUC by ~0.01 from the original. Though the fact that it did diminish the AUC somewhat shows it has enough significance to explain its inclusion in the original set.

Each model was then trained and validated using the expanded set of now 6 features. Additionally, two voting schemes were tested to see if combining the classifiers could improve performance. One used majority or "hard" voting, where each classifier gets 1 vote for the class it would choose and the class voted for the most is used as the predicted class.(Halteren et al. 2001) The other used probability or "soft" voting, where each classifier gives a prediction probability for each class then the class with the highest average probability across the classifiers is selected.(Halteren et al. 2001) The ROC curves were then generated for each model using the new feature set (except for the hard voting model since it does not give a probability score). Each model's performance is as follows:

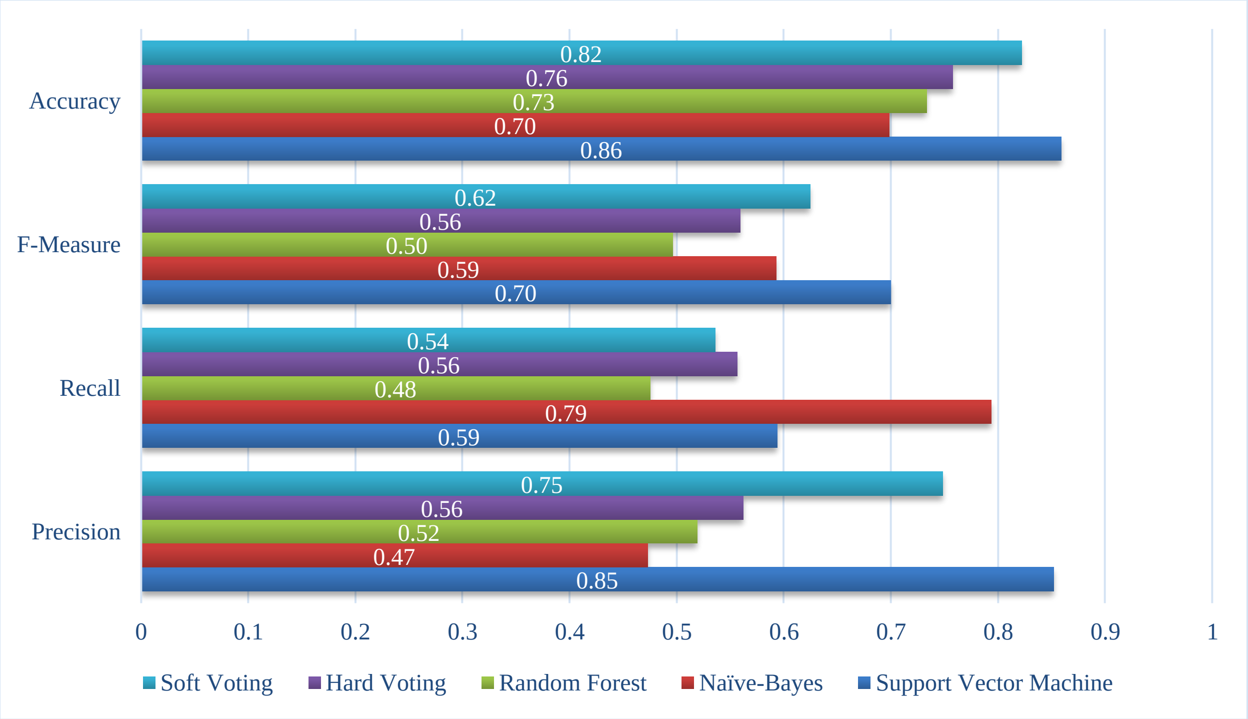

Figure 7. Classifier Performance Comparison with SVM-only Selected Features. The average performance metrics for the classifiers across the 100 experiments. The Activating class was considered the positive class in the calculation of Recall and Precision.

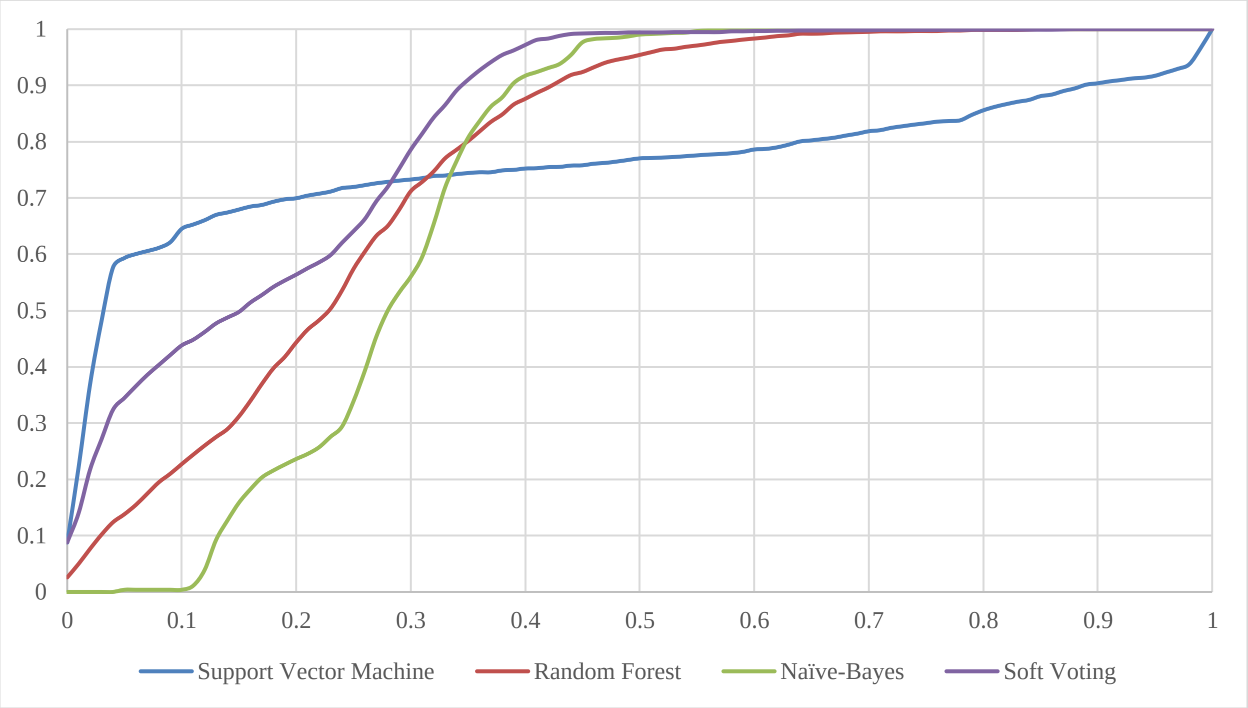

Figure 8. Classifiers' ROC Curves with SVM-only Selected Features: The ROC curve for each of the 3 highest performing classifiers being compared.

Table 6. AUC for Figure 8

Classifier	AUC
SVM	0.760272597
RF	0.744972771
NB	0.721502771
Soft Voting	0.827519307

The new feature set resulted in an across-the-board improvement for the Support Vector Machine. Figure 7 shows, the SVM performed better across the board with the expanded set of features. With respect to the Activating class its Precision increased from 0.80 to 0.85, its Recall from 0.50 to 0.59, and the resulting F-Measure from 0.61 to 0.70. Finally, its Accuracy increased from 0.83 to 0.86. With this improvement, the SVM now performs better than the other classifiers across all of the metrics, whereas with the set of only three features it had a lower Activating Recall than the Random Forest and Naïve-Bayes models. Interestingly, the F-Measure for the other two classifiers increased as well, although the Naïve-Bayes model gained recall but lost precision and accuracy and the Random Forest lost recall but gained precision.(Fig 7) Neither voting scheme improved the performance according to the metrics in Figure 7. Soft voting was the best, with the second highest overall Accuracy, F-Measure, and Precision, though it did not out-perform the SVM in any one of the metrics in Figure 7.

In the ROC curve, we can clearly see the SVM outperforming the other two base classifiers in its ability to detect Activating mutations without a high degree of false positives. Interestingly however, the Random Forest and Naïve-Bayes classifiers do both exceed the SVM's true positive detection capability, however this comes at the cost of greatly increased false positives. Neither model surpasses the SVM until they have a false positive rate greater than 30%, well above that achieved by the SVM at almost the same true positive rate.(Fig 8) However due to their improved performance later in the curve, the AUC for the Random Forest and Naïve-Bayes classifiers are only ~0.015 and ~0.03 lower respectively than that of the SVM. Interestingly, by combining all 3 classifiers using soft voting we don't lose too much performance at the lower end of the false positive rates, while improving the true positive rate later in the curve to higher than even the Random Forest or Naïve-Bayes. This yields an AUC for the soft voting model of 0.82, ~0.06 higher than the SVM.(Table 6) So while the soft voting strategy performed worse in our validation metrics than the SVM, it did perform better in the ROC curve, showing it to be the most effective strategy for classification of the Activating class.

In small datasets such as ours there can be concern that the data is not representative enough of the full problem space to constitute a valid sample for model training/validation. To ease this concern, we generated a learning curve using the set of 6 features and our highest performing classifiers. A learning curve is a method for determining the validity of a dataset as a representative sample.(Perlich 2010) It is created by taking a series of subsample of varying sizes of the training data and using them to train/validate one or more models. With the sample size on the x-axis and the model accuracy on the y-axis, the resulting curve depicts how much adding data improves your model. Ideally, the curve will have a high positive slope as it initially increases its training set size from zero, but then this slope will greatly decrease as more data is added and the actual dataset size is approached. The implication is should the curve level-off as it approaches our dataset size then out dataset is a sufficient sample of the domain since adding anymore data yields only a marginal improvement in accuracy. To calculate ours, cross-validation was employed with each model being trained and validated on varying size subsamples of each fold. This was repeated 100 times with unique partitions and the results were averaged. It is depicted as follows:

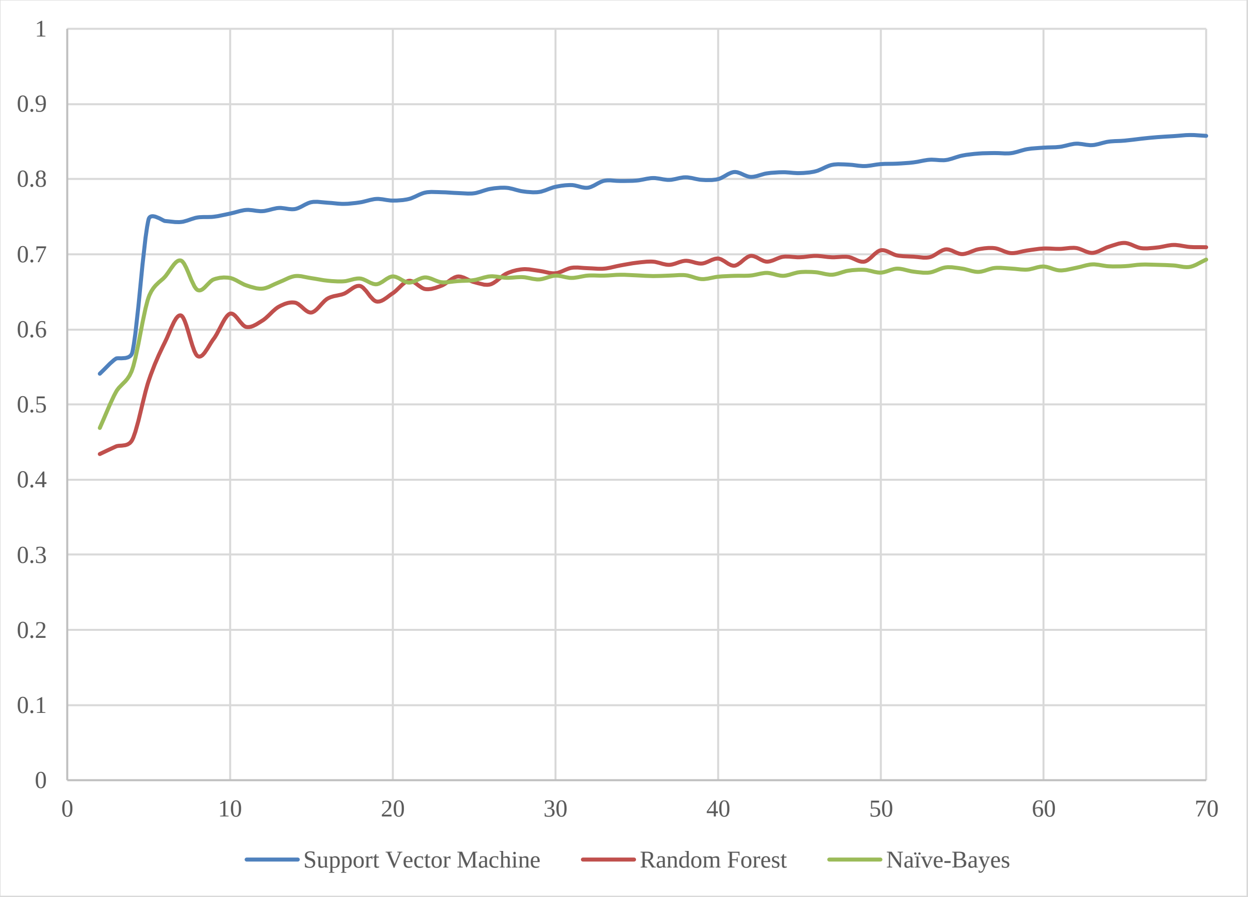

Fig 9. Learning Curve. The Accuracy of each model when trained on a given sample size. The accuracy was determined using cross validation.

As shown in Figure 9, each model undergoes a substantial increase in accuracy as data is added to the smaller subsamples. The accuracy gradually levels off and the rate at which accuracy increased was reduced. However, there is still a moderate improvement as the sample size increases. This curve indicates that while our sample is approaching the ideal sample size additional data would still prove valuable in the training/validation of our models.

After exploring the data and our ability to model it, we next took a set of previously unclassified mutations and evaluated them with the SVM using the six features that yielded the highest performance. The evaluation was done by randomly partitioning the training set into 10 sets of roughly equal size (in a similar manner as cross-fold validation). For each partition, the SVM was trained then voted for the classification of each mutant in the unknown set. If a majority of the partitions voted that an unknown mutant was Activating, it received a score of 1 for that experiment, if more voted it was Non-Activating it received a score of 0 and if equal number voted for either then it received a score of 0.5. These scores were summed across 100 different random splits to generate the predicted likelihood of an unknown mutation being Activating (e.g. a score of 100 being the most likely to be Activating, 0 the least.) We compared these quantitative results from our models to our own qualitative predictions of the potential class for each mutant. A selected sample of the results is as follows:

Table 7. Classifier Predictions for Previously Unclassified Mutations: Table of selected resulting scores from unknown mutant classification as well as our own predictions of the mutant class based on qualitative examination of the mutation.

Variant	SVM Score	Qualitative Prediction	Comments
A859T	100	Activating	A859 is at the activation loop helix right under the C-helix. I expect A859T to destabilize the helix and activate the kinase.
H773Y	100	Non-Activating	This is NPH-His. I expect the mutation to be inactive.
I821T	100	Non-Activating	I821 locates at the E-helix and interact with TK-network residues. I expect threonine to destabilize the kinase core.

L718P	100	Non-Activating	This will be inactivating at Pro will disrupt the beta1 strand.
L747S	100	Activating	L747 is at the end of beta3 strand and interact with the short helix at the A-loop. Mutation to a serine might activate the kinase.
L858A	100	Activating	This is possible at L858A will destabilize the activation loop, although to a lesser extent compared to L858R.
L861E	100	Activating	This could be activating. L861E will clearly destabilize the inactive state of EGFR.
L862Q	100	Activating	L862Q could destabilize the activation loop, thereby activates the kinase.
R776L	100	Activating	R776L will disrupt the auto-inhibitory interaction mediated by R776 **(PMID: 26101090**).
R776S	100	Activating	R776S will disrupt the auto-inhibitory interaction mediated by R776 **(PMID: 26101090**).
V834A	100	Non-Activating	V834 is at the beginning of beta7 strand and packs in a hydrophobic core. I expect alanine to destabilize the protein.
L858G	81	Activating	I expect Glycine will destabilize the protein in the inactive state.
A743P	0	Non-Activating	A743 is at the beta3 strand before KE-Lys. Proline will terminate the strand and inactivate the kinase.
A743T	0	Non-Activating	Same as A743P
A743V	0	Non-Activating	Same as A743T
A763D	0	Non-Activating	A763 is at the C-helix and mediate hydrophobic interaction with the N-lobe. I expect A763D to interfere C-helix position and inactivate the kinase.
A822P	0	Non-Activating	A822 is in the middle of E-helix. A proline will disrupt the helix and affect kinase folding.
A859D	0	Activating	A859 is at the activation loop helix under the C-helix. Mutation it to an Asp will destabilize the inactive state of EGFR.
C775Y	0	Non-Activating	C775 is at the alphaC-beta4 loop and is buried inside the kinase core. Tyr will completely destabilize the alphaC-beta4 loop.
D770N	0	Non-Activating	I have experimental data for this mutation. Its activity is comparable to the WT EGFR. Therefore, the prediction here is correct.
D837G	0	Non-Activating	D837 is the HRD-Asp. This mutation should be inactivating.
D837N	0	Non-Activating	Same at D837G
D855N	0	Non-Activating	D855 is catalytic Asp. Mutation of it will definitely inactivate the kinase.
E884K	0	Non-Activating	E884 is the APE-Glu (EGFR has a ALE). The mutation should be inactive.
F856Y	0	Non-Activating	F856 is the DFG-Phe. Should be non-activating.
G721W	0	Non-Activating	Trp is too bulky here and will affect ATP binding. I expect the mutation to be inactive.
G857E	0	Non-Activating	G857 is the DFG-Gly. I expect this mutation to be non-activating.
G857V	0	Non-Activating	Same as G857E
G873E	0	Non-Activating	G873E is at the activation loop. I expect the mutation impact to be neutral.
G873Q	0	Non-Activating	Same as G873E.

G917A	0	Non-Activating	G917 is in the alphaF-alphaG loop. Unknown mutation impact.
G917R	0	Non-Activating	Same as G917A.
H773L	0	Non-Activating	H773 is the HPN-His. Mutation of the histidine is expect to inactivate the kinase.
H773P	0	Non-Activating	Same as H773L
H773R	0	Non-Activating	Same as H773R
I759N	0	Non-Activating	I759 is in the C-helix. I expect this mutation will affect the correct position of C-helix in the active state.
I759V	0	Non-Activating	Valie is very similar to isoleucine. Unknown mutation impact.
I780S	0	Non-Activating	I780 is in the beta4 strand and makes hydrophobic interaction with C-helix. I expect this mutation to destabilize the interaction between C-helix and kinase N-lobe.
I780T	0	Non-Activating	Same as I780S
K745R	0	Non-Activating	K745 is the KE-Lys. The mutation should be inactive.
L782N	0	Non-Activating	L782 interacts with the C-helix through hydrophobic interactions. Asparagine will destabilize such interaction and inactivate the kinase.
L815P	0	Non-Activating	L815 is in the E-helix. A proline will probably determinate the E-helix and affect kinase folding.
L844P	0	Non-Activating	L844 is in the beta7 strand and interact with ATP molecule. Proline will disrupt the secondary structure here and inactivate the kinase.
L858K	0	Activating	I expect lysine will be similar to arginine and activate the kinase at the position.
L858P	0	Activating	Proline will also destabilize the 3/10 helix and activates the kinase.
L861G	0	Activating	Glycine will create a void here and destabilize the inactive state.
L862R	0	Activating	Arginine will probably destabilize the 3/10 helix as well.
N842D	0	Non-Activating	N842 is the HRD+5 Asn. Mutation to Asp will probably inactivate the kinase.
N842H	0	Non-Activating	Same as N842D
N842S	0	Non-Activating	Same as N842D
P772H	0	Non-Activating	P772 is the NPH-Pro. I expect a histidine here will inactivate the kinase.
P772R	0	Non-Activating	Same as P772H
P772S	0	Non-Activating	Same as P772H

CHAPTER 4

CONCLUSION

Study Limitations and Conclusion

The limitations of our study are primarily due to the lack of data as compared to the extreme complexity of this domain's feature space. For instance, we had to perform feature selection upon the same dataset as we performed the training and cross-validation of our model. Though we employed comprehensive methods to combat the potential of over-fitting and information leakage due to this, ideally were there sufficient data we would keep entirely separate sets for feature selection and model. We also believe that some of the features found to have no significant contribution for classification in this dataset could provide useful insight should we have a dataset large enough to accommodate them. This issue stems from the relatively small size of our dataset as compared to the high degree of variability in mutations that could occur. The complexity of EGFR kinase domain (with a total of 317 residues in our model) leads to the potential of 6023 distinct point mutations, and finding a ground truth for each is incredibly time consuming (hence the purpose and importance of our research). Across each of these mutations there is a high dimension of potential features, each with a large degree of variability across their respective dimension. Within our own dataset there are additional features we potentially could have calculated, further increasing the problem's complexity. This high degree of variability of data points within such a complex feature space makes finding a generalizable predictive model challenging.

Despite these limitations, we proposed an effective feature selection technique, which thoroughly examined the feature space while minimizing the risk of overfitting and data leakage. This scheme found two feature set that were consistently chosen across a substantial variety of data subsamples/permutations. One of the feature sets was a super set of the other, further reinforcing the approaches effectiveness in finding the globally meaningful features. Finally, a variety of model were validated with a Support Vector Machine using a Gaussian radial basis function kernel performing the best. It achieved an Accuracy of 86% and an F-Measure of 0.7. This model was rigorously validated in such a way as to negate the possibility of its performance being biased by overfitting and data leakage. It was then combined with the next two highest performing models, Random Forest and Gaussian Naïve-Bayes in a weighted probability voting scheme which achieved an ROC AUC score of ~0.83.

While compiling our dataset, we also noticed a lack of a standardized method of assessing mutational impact. Our dataset requires the mutant kinase to be assayed directly for their auto-phosphorylation activity. Classifiers trained on databases such as COSMIC (Forbes et al. 2008) would define mutational impact as simply appearing in a cancer sequencing. A recently developed high-throughput assay assesses mutants by increased cell proliferation (Kohsaka et al. 2017). Further complicating the issue, recent work has shown that the kinase signaling network is capable of flux under aberrant signaling conditions (Wilson 2018). While this issue may never be fully resolved, we believe it is important to maintain a consistent and concrete definition within datasets.

Despite the issues associated with a small dataset, our high level of curation allowed us to accurately analyze the importance of features in predicting disease-related mutations. We narrowed our dataset to kinase activating mutations in EGFR and performed extensive feature selected followed by training of machine learning models. While widely-used features such as structural properties and, in the case of SVM, sequence conservation were indeed found to be good predictors of EGFR activation. Our analysis demonstrates: 1) conformation is an important factor when calculating structural features, 2) the specific level of sequence conservation plays a large role in its importance, and 3) EGFR-specific features have high predictive power. We proposed biological explanations as to why these features are important in predicting EGFR activity. We also applied our models to unclassified data to attain meaningful, discussion worthy result. We have demonstrated the existence of meaningful, though not yet fully identifiable, trends within this data that merit the continuation of this line of research.

REFERENCES

1. Adzhubei IA, Schmidt S, Peshkin L, Ramensky VE, Gerasimova A, Bork P, et al. A method and server for predicting damaging missense mutations. Nature Methods. 2010;7: 248–249. doi:10.1038/nmeth0410-248
2. Arlot S, Celisse A. A survey of cross-validation procedures for model selection. Statistics Surveys. 2010;4: 40–79. doi:10.1214/09-ss054
3. Berliner N, Teyra J, Çolak R, Lopez SG, Kim PM. Combining Structural Modeling with Ensemble Machine Learning to Accurately Predict Protein Fold Stability and Binding Affinity Effects upon Mutation. PLoS ONE. 2014;9. doi:10.1371/journal.pone.0107353
4. Blum AL, Langley P. Selection of relevant features and examples in machine learning. Artificial Intelligence. 1997;97: 245–271. doi:10.1016/s0004-3702(97)00063-5
5. Breiman L. Random Forests. Machine Learning. 2001;45: 5–32. doi:10.1023/a:1010933404324
6. Cario CL, Witte JS. Orchid: a novel management, annotation and machine learning framework for analyzing cancer mutations. Bioinformatics. 2017;34: 936–942. doi:10.1093/bioinformatics/btx709
7. Choi SH, Mendrola JM, Lemmon MA. EGF-independent activation of cell-surface EGF receptors harboring mutations found in gefitinib-sensitive lung cancer. Oncogene. 2006;26: 1567–1576. doi:10.1038/sj.onc.1209957
8. Cover T, Hart P. Nearest neighbor pattern classification. IEEE Transactions on Information Theory. 1967;13: 21–27. doi:10.1109/tit.1967.1053964

9. Dehouck Y, Kwasigroch JM, Rooman M, Gilis D. BeAtMuSiC: prediction of changes in protein–protein binding affinity on mutations. Nucleic Acids Research. 2013;41. doi:10.1093/nar/gkt450

10. Denœux T. A k-Nearest Neighbor Classification Rule Based on Dempster-Shafer Theory. Classic Works of the Dempster-Shafer Theory of Belief Functions Studies in Fuzziness and Soft Computing. 1995;: 737–760. doi:10.1007/978-3-540-44792-4_29

11. Dreiseitl S, Ohno-Machado L. Logistic regression and artificial neural network classification models: a methodology review. Journal of Biomedical Informatics. 2002;35: 352–359. doi:10.1016/s1532-0464(03)00034-0

12. Fu Y, Liu Z, Lou S, Bedford J, Mu XJ, Yip KY, et al. FunSeq2: a framework for prioritizing noncoding regulatory variants in cancer. Genome Biology. 2014;15. doi:10.1186/s13059-014-0480-5

13. Giollo M, Martin AJ, Walsh I, Ferrari C, Tosatto SC. NeEMO: a method using residue interaction networks to improve prediction of protein stability upon mutation. BMC Genomics. 2014;15. doi:10.1186/1471-2164-15-s4-s7

14. Gnad F, Baucom A, Mukhyala K, Manning G, Zhang Z. Assessment of computational methods for predicting the effects of missense mutations in human cancers. BMC Genomics. 2013;14. doi:10.1186/1471-2164-14-S3-S7

15. Gold MS, Bentler PM. Treatments of Missing Data: A Monte Carlo Comparison of RBHDI, Iterative Stochastic Regression Imputation, and Expectation-Maximization. Structural Equation Modeling: A Multidisciplinary Journal. 2000;7: 319–355. doi:10.1207/s15328007sem0703_1

16. Gustavo E. A. P. A. Batista, Prati RC, Monard MC. A study of the behavior of several methods for balancing machine learning training data. ACM SIGKDD Explorations Newsletter. 2004;6: 20. doi:10.1145/1007730.1007735

17. Hall M. Correlation-based Feature Selection for Machine Learning. 1999;

18. Halteren HV, Zavrel J, Daelemans W. Improving Accuracy in Word Class Tagging through the Combination of Machine Learning Systems. Computational Linguistics. 2001;27: 199–229. doi:10.1162/089120101750300508

19. Hanley JA, Mcneil BJ. The meaning and use of the area under a receiver operating characteristic (ROC) curve. Radiology. 1982;143: 29–36. doi:10.1148/radiology.143.1.7063747

20. James KA, Verkhivker GM. Structure-Based Network Analysis of Activation Mechanisms in the ErbB Family of Receptor Tyrosine Kinases: The Regulatory Spine Residues Are Global Mediators of Structural Stability and Allosteric Interactions. PLoS ONE. 2014;9. doi:10.1371/journal.pone.0113488

21. Jura N, Endres NF, Engel K, Deindl S, Das R, Lamers MH, et al. Mechanism for Activation of the EGF Receptor Catalytic Domain by the Juxtamembrane Segment. Cell. 2009;137: 1293–1307. doi:10.1016/j.cell.2009.04.025

22. Kancha RK, Bubnoff NV, Peschel C, Duyster J. Functional Analysis of Epidermal Growth Factor Receptor (EGFR) Mutations and Potential Implications for EGFR Targeted Therapy. Clinical Cancer Research. 2009;15: 460–467. doi:10.1158/1078-0432.ccr-08-1757

23. Kohsaka S, Nagano M, Ueno T, Suehara Y, Hayashi T, Shimada N, et al. A method of high-throughput functional evaluation of EGFR gene variants of unknown significance in cancer. Science Translational Medicine. 2017;9. doi:10.1126/scitranslmed.aan6566

24. Lavoie H, Li JJ, Thevakumaran N, Therrien M, Sicheri F. Dimerization-induced allostery in protein kinase regulation. Trends in Biochemical Sciences. 2014;39: 475–486. doi:10.1016/j.tibs.2014.08.004

25. Lewis DD. Naive (Bayes) at forty: The independence assumption in information retrieval. Machine Learning: ECML-98 Lecture Notes in Computer Science. 1998;: 4–15. doi:10.1007/bfb0026666

26. Lou W, Wang X, Chen F, Chen Y, Jiang B, Zhang H. Sequence Based Prediction of DNA-Binding Proteins Based on Hybrid Feature Selection Using Random Forest and Gaussian Naïve Bayes. PLoS ONE. 2014;9. doi:10.1371/journal.pone.0086703

27. Martens HA, Dardenne P. Validation and verification of regression in small data sets. Chemometrics and Intelligent Laboratory Systems. 1998;44: 99–121. doi:10.1016/s0169-7439(98)00167-1

28. Mcskimming DI, Dastgheib S, Talevich E, Narayanan A, Katiyar S, Taylor SS, et al. ProKinO: A Unified Resource for Mining the Cancer Kinome. Human Mutation. 2015;36: 175–186. doi:10.1002/humu.22726

29. Niroula A, Vihinen M. Predicting Severity of Disease-Causing Variants. Human Mutation. 2017;38: 357–364. doi:10.1002/humu.23173

30. Pan Y, Liu D, Deng L. Accurate prediction of functional effects for variants by combining gradient tree boosting with optimal neighborhood properties. Plos One. 2017;12. doi:10.1371/journal.pone.0179314

31. Patankar B, Chavda V. Effect of Feature Selection Using Best First Search on the Performance of Classification. International Journal of Scientific Research in Science and Technology. 2016;2.

32. Pedregosa *et al.*, Scikit-learn: Machine Learning in Python JMLR 12, pp. 2825-2830, 2011

33. Perlich C. Learning Curves in Machine Learning. SpringerReference. 2010; doi:10.1007/springerreference_179164

34. Pons T, Vazquez M, Matey-Hernandez ML, Brunak S, Valencia A, Izarzugaza JM. KinMutRF: a random forest classifier of sequence variants in the human protein kinase superfamily. BMC Genomics. 2016;17. doi:10.1186/s12864-016-2723-1

35. Pucci F, Bourgeas R, Rooman M. Predicting protein thermal stability changes upon point mutations using statistical potentials: Introducing HoTMuSiC. Scientific Reports. 2016;6. doi:10.1038/srep23257

36. Puente XS, Pinyol M, Quesada V, Conde L, Ordóñez GR, Villamor N, et al. Whole-genome sequencing identifies recurrent mutations in chronic lymphocyticleukaemia. Nature. 2011;475: 101–105. doi:10.1038/nature10113

37. Quan L, Lv Q, Zhang Y. STRUM: structure-based prediction of protein stability changes upon single-point mutation. Bioinformatics. 2016;32: 2936–2946. doi:10.1093/bioinformatics/btw361

38. Ruan Z, Kannan N. Mechanistic Insights into R776H Mediated Activation of Epidermal Growth Factor Receptor Kinase. Biochemistry. 2015;54: 4216–4225. doi:10.1021/acs.biochem.5b00444

39. Ruan Z, Katiyar S, Kannan N. Computational and Experimental Characterization of Patient Derived Mutations Reveal an Unusual Mode of Regulatory Spine Assembly and Drug Sensitivity in EGFR Kinase. Biochemistry. 2016;56: 22–32. doi:10.1021/acs.biochem.6b00572

40. Ruggieri S. Efficient C4.5 [classification algorithm]. IEEE Transactions on Knowledge and Data Engineering. 2002;14: 438–444. doi:10.1109/69.991727

41. Scholkopf B, Sung K-K, Burges C, Girosi F, Niyogi P, Poggio T, et al. Comparing support vector machines with Gaussian kernels to radial basis function classifiers. IEEE Transactions on Signal Processing. 1997;45: 2758–2765. doi:10.1109/78.650102

42. Shanker M, Hu M, Hung M. Effect of data standardization on neural network training. Omega. 1996;24: 385–397. doi:10.1016/0305-0483(96)00010-2

43. Shi Z, Moult J. Structural and Functional Impact of Cancer-Related Missense Somatic Mutations. Journal of Molecular Biology. 2011;413: 495–512. doi:10.1016/j.jmb.2011.06.046

44. Sokolova M, Japkowicz N, Szpakowicz S. Beyond Accuracy, F-Score and ROC: A Family of Discriminant Measures for Performance Evaluation. Lecture Notes in Computer Science AI 2006: Advances in Artificial Intelligence. 2006;: 1015–1021. doi:10.1007/11941439_114

45. U M, Talevich E, Katiyar S, Rasheed K, Kannan N. Prediction and Prioritization of Rare Oncogenic Mutations in the Cancer Kinome Using Novel Features and Multiple Classifiers. PLoS Computational Biology. 2014;10. doi:10.1371/journal.pcbi.1003545

46. Vaser R, Adusumalli S, Leng SN, Sikic M, Ng PC. SIFT missense predictions for genomes. Nature Protocols. 2015;11: 1–9. doi:10.1038/nprot.2015.123

47. Velankar et al., Nucleic Acids Research 41, D483. 2013

48. Wagih O, Reimand J, Bader GD. MIMP: predicting the impact of mutations on kinase-substrate phosphorylation. Nature Methods. 2015;12: 531–533. doi:10.1038/nmeth.3396

49. Wilson DC, Smyth B, Sullivan DO. Sparsity Reduction in Collaborative Recommendation: A Case-Based Approach. International Journal of Pattern Recognition and Artificial Intelligence. 2003;17: 863–884. doi:10.1142/s0218001403002678

50. Wilson LJ, Linley A, Hammond DE, Hood FE, Coulson JM, Macewan DJ, et al. New Perspectives, Opportunities, and Challenges in Exploring the Human Protein Kinome. Cancer Research. 2017;78: 15–29. doi:10.1158/0008-5472.can-17-2291

51. Yates CM, Filippis I, Kelley LA, Sternberg MJ. SuSPect: Enhanced Prediction of Single Amino Acid Variant (SAV) Phenotype Using Network Features. Journal of Molecular Biology. 2014;426: 2692–2701. doi:10.1016/j.jmb.2014.04.026

52. Ye J, Wang T. Regularized discriminant analysis for high dimensional, low sample size data. Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining - KDD 06. 2006; doi:10.1145/1150402.1150453

53. Yi S, Lin S, Li Y, Zhao W, Mills GB, Sahni N. Functional variomics and network perturbation: connecting genotype to phenotype in cancer. Nature Reviews Genetics. 2017;18: 395–410. doi:10.1038/nrg.2017.8

54. Zhang X, Gureasko J, Shen K, Cole PA, Kuriyan J. An Allosteric Mechanism for Activation of the Kinase Domain of Epidermal Growth Factor Receptor. Cell. 2006;125: 1137–1149. doi:10.1016/j.cell.2006.05.013